ANO II - NÚMERO 13
Cr$ 900,00

Interface

Hardware - Software - Computadores - Mini - Micro - Teleprocessamento

MANAUS, SANTARÉM, RIO BRANCO, ALTAMIRA, BOAVISTA, MACAPÁ E PORTO VELHO — Cr$ 1.100,00

Seleção de Software Aplicativo

ASSEMBLER NO TK-82C

Jorge Gallo

68.000
O Poderoso Microprocessador de 16 Bits

SINTA NOS DEDOS ESTA NOVA CONQUISTA

Já não é preciso escolher. Agora você tem o microcomputador DGT-1000, com design moderno e novas incorporações que lhe conduzirão à decisão certa.
O DGT-1000 é modular e dependendo da sua necessidade ele se expande até um grande sistema.
A DIGITUS comercializa a interface printer, usada para conectar o DGT-1000 à impressora.
A interface é tipo paralelo "CENTRONICS" cujo programa de acesso já está incluído na ROM do computador e as instruções são do próprio BASIC.

DGP/M

DIGITUS
Rua Gávea, 150 Belo Horizonte
Tel.: (031) 332-8300 Telex: 3352

Aracajú (079) 224-1310/224-6111. **Brasília** (061) 242-6344/226-8701/226-9201/273-2128/225-4534/224-3505/226-2374/248-5030/561-3307/248-6321. **Belo Horizonte** (031) 222-7889/223-6947/226-6336/226-5734/226-9078/225-3305/225-2469. **Belém** (091) 224-9988. **Campinas** (0192) 32-4155/32-3810. **Campo Grande** (067) 383-1068/382-6587. **Cuiabá** (065) 322-9713. **Curitiba** (041) 244-1405/224-0544/232-1750/243-1731. Fortaleza (085) 227-5878/244-0544/244-4691/226-4922/224-7864/231-4822. **Florianópolis** (0482) 23-1039. **Frederico Westphalen** (055) 334-1550/334-1672. **Goiânia** (062) 225-8598/223-1122. **João Pessoa** (083) 221-6743. **Maceió** (082) 223-3979. **Niterói** (021) 714-0112/ 722-6791/717-1570. **Novo Hamburgo** (0512) 93-4721. **Natal** (084) 222-3212. **Montes Claros** (038) 221-8212. **Ouro Preto** (031) 551-1933. **Porto Alegre** (0512) 22-9782/26-8468/ 26-8246/40-1998/21-4189. **Recife** (081) 326-9969/326-9318/ 222-4714. **Ribeirão Preto** (016) 636-0586. **Rio de Janeiro** (021) 264-0143/262-8737/322-4166/242-2752/ 221-8282/295-8194/267-8291/252-2050/252-4080/228-0734/248-8159/284-5649/247-7842/222-6088/259-1516/288-2650/267-1093/267-1443/252-9057/263-1241/ 391-8965/286-4849/591-3297/249-3166/267-1093/252-9420. **Salvador** (071) 247-4936/245-6198/243-2684/242-9394/248-6666/235-4184. **Santa Maria** (055) 221-7120. **São Paulo** (011) 222-1511/283-0596/852-2958/282-2105/258-3954/227-6100/ 227-4433/280-2322/212-9004/210-0187/61-4049/61-0949/258-7311/814-3663/231-3922. **Taubaté** (0122) 32-9807. **Poços de Caldas** (035) 721-5810.

conteúdo

Interface

matérias

INFORMÁTICA/83
Quando a Fantasia se Torna Realidade 6
Por Barbara Hartz

68000 – O Poderoso Microprocessador de 16 Bits (1ª Parte) 12

A arquitetura do mais poderoso microprocessador de 16 bits

Por Cesar da Costa

Curso Microprocessador Z-80 (12ª Lição) 17

Instruções lógicas e aritméticas de 8 bits

Por André Gil Rubens e Ney Acyr de Oliveira (A & N – Consultoria em Hardware e Software de Microcomputadores)

1º Encontro Regional dos Usuários de Microcomputadores 36

Seleção de Software Aplicativo

Palestra de Tarcisio N. da Cunha

Filia: Um Sistema de Programação para Microcomputadores (6ª Parte) 42

Compiladores do sistema

Por Rildo Pragana

MUMPS: Características e Comparações com Outros Sistemas de Programação (Parte Final) 46

Manipulação de dados DBTG, Banco de Dados Relacionais, comparações com COBOL, FORTRAN, PL/1, APL e BASIC

departamentos

Notas do Editor 2
Informe 4
Consultoria 11
Cartas 16
Novos Produtos 22
Calendário 24
Livros 45

seções

Educação/Tecnologia 52

A Atual Política de Informática Brasileira

Por Reinaldo de Freitas Pires de Moraes

O Sistema Operacional UNIX (2ª Parte) 54

A linguagem "C"

Por Ivan Costa e Mônica Fróes Peixoto

SUPLEMENTO DO PEQUENO USUÁRIO DE MICROCOMPUTADORES

Curso de Programação Basic (4ª Lição) 25

Entrada de dados, saída de dados, programas e exercícios

Por José Arthur da Rocha

Programas Aplicativos 29

Contas a pagar/receber no TK82-C, CP200, TK85, SINCLAIR e similares

Por Francisco José D. Figueiredo

Dicas de Hardware 32

Implementação da função SLOW no NEZ-8000

Por Cesar da Costa

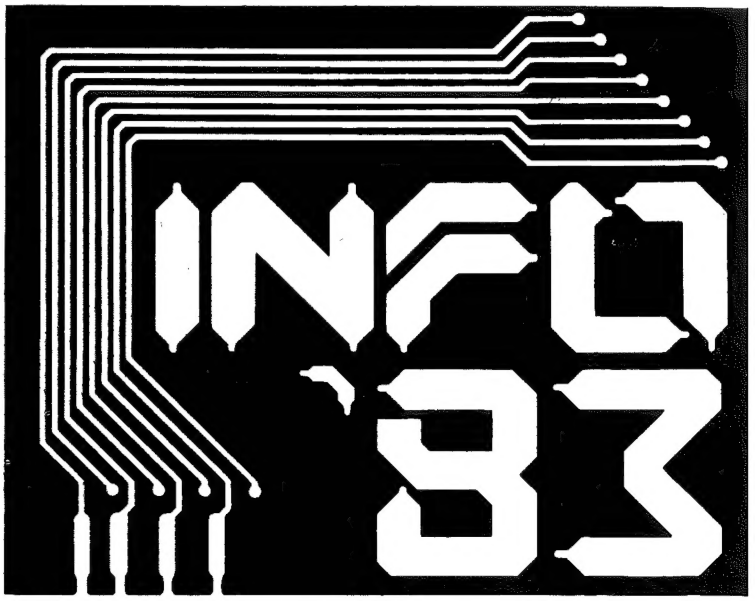
Dicas de Software 34

Assembler no TK82-C

Por Mauricio Costa Reis

I Comunidade de Projetistas de Hardware e Software – COPHAS 39

Glossário de Termos Técnicos 40



DIREÇÃO:
Naila Glória O. Côrtes
Cesar da Costa

EDITORIA TÉCNICA:
Cesar da Costa
Naila Glória O. Côrtes

CONSULTORIA TÉCNICA:
Ivan Costa
Tarcísio Neves da Cunha
Manuel Lois Anido
José Arthur da Rocha
André Gil Rubens
Ney Acyr de Oliveira
Cesar de Araújo Lima

PRODUÇÃO/DIAGRAMAÇÃO:
Nilton Luiz Côrtes

ARTE FINALIZAÇÃO:
Jorge Gatto
Sandra Maria Pinto dos Santos

REDAÇÃO/REVISÃO:
Nelma Bernadete C. de Castro

DIREÇÃO COMERCIAL:
José Augusto R. Pereira

Publicidade/RJ:
Est. do Tindiba, 2380
Tel.: (021) 392-8965

Contatos:
Roberto Passeri
José Augusto Nunes Rodrigues
Tel.: (021) 234-3599

Publicidade/SP:
Av. Paulista, 1159 — Conj. 801
Tel.: (011) 284-8384

Gerente:
Araci Monteiro Pedreira

ASSINATURAS:

Supervisor/RJ:
Edson Teixeira Pedroso
Tel.: (021) 234-3599

Supervisor/SP:
Luiz Henrique C. Guimarães
Av. Paulista, 1159 — Conj. 801
Tel.: (011) 284-8384

Supervisor/BH:
Ildo Idolar Peres

Preços:
Assinaturas/Brasil: Cr$ 10.800,00 (1 ano)
Cr$ 19.800,00 (2 anos)

Assinaturas/Exterior: US$ 50,00 (1 ano)
US$ 100,00 (2 anos)

Serviço Centralizado de Atendimento ao Assinante/Leitor:
Tel.: 234-3599
Caixa Postal 3954 — CEP 20001

TIRAGEM:
35.000 exemplares

DISTRIBUIÇÃO:
Abril S/A Cultural e Industrial — Divisão Distribuição (exclusivo para todo o Brasil)

COMPOSIÇÃO:
Studio Moraes — Tel.: 256-0939

FOTOLITOS/IMPRESSÃO:
Gráfica LORD

Interface é editada pela Prodigt — Processamento, Tecnologia e Comunicação Ltda.
Administração e Redação:
Estrada do Tindiba, 2380 — Jacarepaguá — Rio de Janeiro — RJ — CEP 22700 — Tel.: 392-8965.
Interface não aceita matéria redacional paga. Todos os direitos de reprodução desta publicação estão reservados, só podendo ser reproduzidos com fins comerciais, mediante autorização prévia.
Os artigos assinados são de responsabilidade do autor, as opiniões emitidas não são necessariamente coincidentes com as da Revista.
Registro no INPI protocolo nº 810965755.

notas do editor

"Os computadores encontram-se entre as realizações humanas mais espantosas e inquietantes, pois ampliam o poder da nossa mente como a tecnologia ampliou a força dos nossos músculos e não sabemos aonde nossas próprias mentes nos levarão por fim . . .

E eles nos ajudarão a todos — não apenas a uns poucos "supertecnocratas" — a pensar mais profundamente sobre nós mesmos e sobre o mundo . . . Pois até mesmo a melhor mente humana acha difícil nutrir, muito menos manipular, mais de umas poucas variáveis de uma vez".

Teriam sido estas assertivas de Alvin Tofler em a "Terceira Onda" uma das causas que moveu milhares de pessoas à III Feira de Informática? Ou o "modismo" teria sido determinante de todo o alvoroço?

O interesse despertado pela Informática é fato observável nos diversos setores profissionais. Ele se insere causando um verdadeiroimpacto e ocasionando uma necessidade premente desses profissionais, dos mais diversos setores, se chegarem à máquina e dela se inteirarem. Os jogos de lazer, em suas diversas modalidades, chegam aos lares nos mais diversos pontos do país. No entanto, podemos perceber o "modismo" arrefecendo e, estamos entrando num momento de seriedade. Parece que chega até nós a era do "profissionalismo" em seus diversos níveis. O interesse legítimo pelo computador está substituindo a simples curiosidade que levou muitos a investirem em uma máquina que não sabiam como utilizar.

O fenômeno microcomputador encarado como "modismo" foi desastroso. Inúmeros empresários compraram equipamentos pela aparência ou porque alguém havia recomendado, constatando posteriormente uma grande inutilidade tendo em vista a circunstância em que o mesmo foi adquirido.

A aquisição de um microcomputador, no entanto, envolve mais variáveis e não pode ser determinada pela aparência ou por uma simples recomendação. Ela deve ser determinada pelo problema ou destino final deste computador ou então se comprará apenas mais um problema e, no caso, ficaremos com os dois. A solução existe mas devemos saber encontrá-la. Uma boa orientação é fundamental para qualquer compra, especialmente aquela que implica em imobilização de capital num momento crítico da economia como é este. A Informática não é um blefe.

Com este alerta, que nos vem a propósito da III Feira de Informática realizada de 17 a 23 de outubro no Parque Anhembi, abrimos esta edição — a primeira do ano II de INTERFACE.

A Feira nos propiciou uma brilhante exposição de máquinas e equipamentos sedutores, APPLES e SINCLAIRS em quantidade. E a reserva de mercado tentando sustentar-se apesar do assédio.

Neste primeiro ano, foram poucas as amenidades; a batalha foi intensa mas aqui estamos surpreendendo as expectativas e ousando a própria superação. O apoio recebido de leitores e fabricantes durante a realização da Feira, quando comemorávamos também o nosso primeiro aniversário, significou uma das poucas amenidades encontradas.

A semente "pegou" e aqui estamos iniciando esta nova etapa, retificando as propostas anteriores e desejando boa leitura com as habituais seções.

Apresentamos uma cobertura sobre o que aconteceu no Anhembi — Quando a Fantasia se torna Realidade; *68000* — O Poderoso Microprocessador de 16 bits; e Assembler no TK-82C. *Ainda, os* Cursos de BASIC e Z-80, Seleção de Software Aplicativo *e* Sistema Operacional Unix.

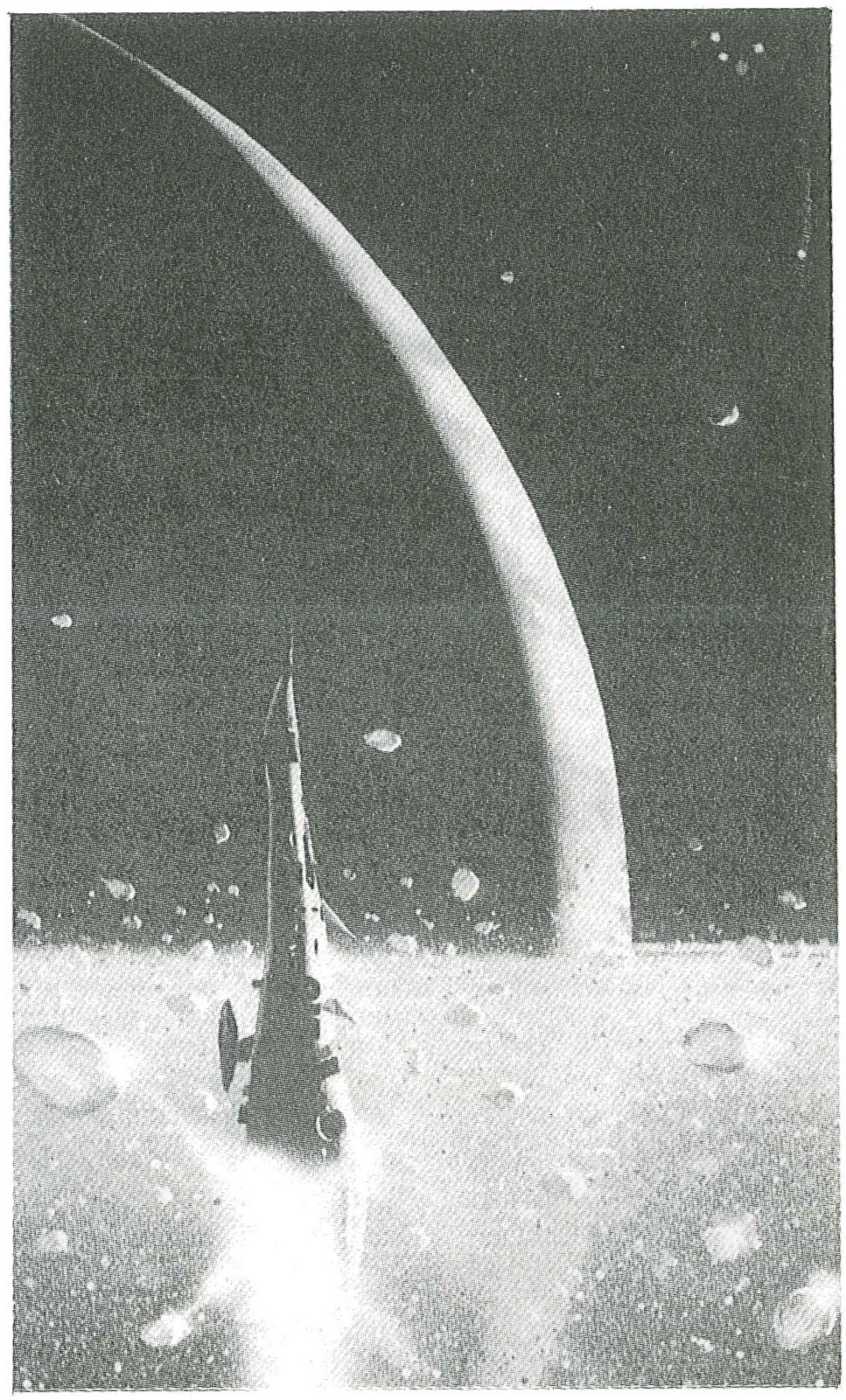
E de presente, Implementação da Função Slow no NEZ-8000.

Ivan Costa

GRANDE PROJETO SINCLAIR*.

Nós comercializamos, você ganha.

Se você é usuário de um dos micros que utilizam lógica Sinclair, participe.



Como participar.

1 - Se você tem:

Programas Inéditos em linguagem de máquina ou Basic, utilitários ou jogos, envie descrição de funcionamento e características.
Projetos Originais em Hardware e deseja ver seu projeto sendo distribuído em nível nacional, mande descrição de funcionamento e características técnicas. Se você já tem um protótipo, informe-nos, também.

2 - Se você quer:

Trabalhar neste grande projeto nacional, como nosso funcionário, trabalhando tanto na parte de Soft quanto em Hardware, envie seu currículo, pretensões salariais, e descrição de sua experiência em lógica Sinclair.

3 - Se você escreve:

Artigos técnicos originais para lógica Sinclair, dicas de programação, textos que você julgue interessante, escreva-nos dando descrições de seu artigo.

4 - Se você deseja:

Receber **gratuitamente** publicação especializada em microcomputadores com lógica Sinclair, simplesmente escreva.

Atenção:

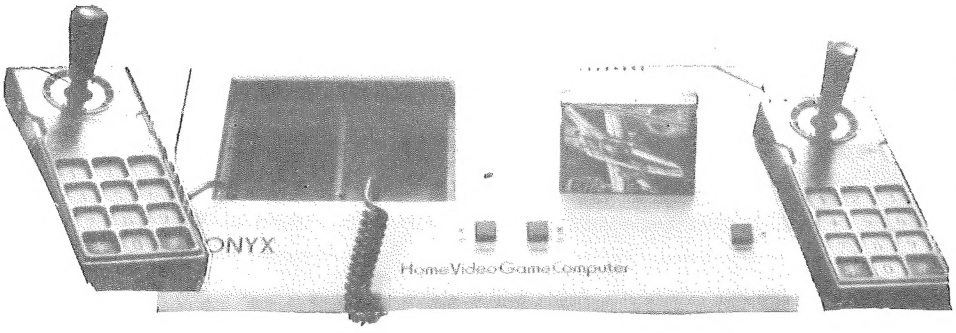
- Os projetos e/ou programas aprovados por nossa equipe serão contatados diretamente para acerto da comercialização. As modalidades de pagamento dos direitos serão informados, na época, aos autores de nosso interesse.
- Todas as cartas serão respondidas, independentemente de aprovação do projeto ou programa.
- Envie toda correspondência para:

GRANDE PROJETO SINCLAIR
Caixa Postal 15185
Rio de Janeiro - RJ

sinclair

* Sinclair é marca registrada da Sinclair Research

informe



ONYX, O SUPER VIDEOGAME, E TK 2000-COLOR – AS GRANDES ESTRELAS DA MICRODIGITAL NA III FEIRA INTERNACIONAL DE INFORMÁTICA

Grande realismo de imagem e som e um avançado design são as características do mais recente lançamento da MICRODIGITAL: ONYX, o SUPER VIDEOGAME.

Sua alta resolução gráfica, 16 cores nitidamente definidas e qualidade técnica são resultantes de trabalhos desenvolvidos durante um ano por engenheiros e técnicos da MICRODIGITAL, preocupados em oferecer ao seu público o mais atraente e inovador videogame.

Acompanhado por duas alavancas de controle (Joysticks), cada uma delas contendo um mini-teclado, este aparelho produz variados efeitos sonoros e confere a impressão de realismo tridimensional a seus programas, tão sofisticados quanto os arcades (fliperamas).

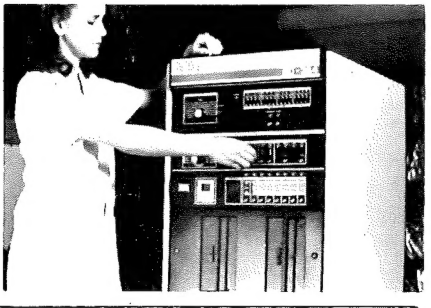
De imediato, estarão à disposição do usuário 20 cartuchos bem elaborados, e, posteriormente, outros serão lançados.

O TK 2000-COLOR foi concebido com o objetivo de servir a pequenas empresas, profissionais liberais e pessoas que gostem de computador. Este lançamento da MICRODIGITAL possibilita o uso de programas nas áreas comerciais, profissionais e recreativas.

O TK 2000-COLOR possui microprocessador 6502, memória de 16K bytes ROM (capacidade de memória operacional) e 64K bytes RAM (capacidade de armazenamento), display gráfico de baixa e alta resolução, unidade de disquete para armazenamento de programas, além de conexão paralela para impressora. Seu teclado profissional, amplo e de fácil manuseio, é composto por 54 teclas.

PROCESSAMENTO TRANSACIONAL

O minicomputador ED-384, um lançamento da série ED-300, é elemento integrante do processamento transacional. Possui 256 KB de memória principal e 4 MB de memória virtual, além de diversos periféricos como: discos, fitas, terminais de vídeo, financeiros e de pontos de venda, etc.



O JÚNIOR DA ITAUTEC

A ITAUTEC está lançando o Júnior, um computador pessoal, filho do Microcomputador ITAUTEC I-7000.

Tal pai, tal filho . . .

O Júnior tem as mesmas características do pai, ou seja, a mesma arquitetura, os mesmos softwares, a mesma compatibilidade com o CP/M e a mesma capacidade de operar como Terminal de Vídeo IBM.

Todos os softwares do I-7000, seja o sistema operacional, o Interpretador Basic, o Redator ou o Calctec, são compatíveis com o Júnior.

O Júnior usa também cartuchos de programas e os mesmos periféricos do *senior,* como por exemplo impressora de 80 ou 100 cps ou *Daisy Wheel.*

O Júnior tem sua capacidade de crescimento garantida pela ITAUTEC; a qualquer momento ele pode crescer e se transformar no I-7000, ou seja, não fica radicalmente limitado em sua capacidade de expansão.

A ITAUTEC pode apresentar um produto nessas condições porque projetou o I-7000; tendo o domínio sobre o projeto, pode oferecer um produto de menor custo, na faixa dos micros de uso pessoal, apresentando as mesmas características de um micro profissional, e que possa evoluir conforme a evolução do usuário.

CABEÇOTE DE LEITURA PARA DISCOS FLEXÍVEIS NACIONAL

No seu esforço para obter uma total nacionalização de seus produtos, a PROLÓGICA acaba de dar um passo importante. Depois de vários estudos, da elaboração de um projeto e dos testes necessários já está fabricando o cabeçote de leitura, um dos componentes usado nos acionadores de disco flexível. Este conjunto que era totalmente importado passa, agora, a ter uma peça fabricada, com grande êxito, aqui mesmo. O projeto do cabeçote de leitura foi elaborado para atender os dois tipos de acionadores, tanto da face simples como dupla, e, assim, o problema de importação ficou totalmente solucionado.

BBS-BIBLIOTECA BRASILEIRA DE SOFTWARE

Está funcionando em São Paulo, e brevemente no Rio de Janeiro, a primeira biblioteca brasileira de software. São quatro mil títulos abrangendo variados assuntos, desde o mais simples jogo até um sofisticado programa para organizar a contabilidade de uma pequena empresa. A idéia da biblioteca nasceu da experiência de três amigos, que atuam na área de computadores há 15 anos. O acervo da BBS é de quatro mil títulos, software adquiridos particularmente pelos próprios sócios do empreendimento e alguns cedidos por softhouses. Os assinantes também poderão colocar os seus software à disposição de outros usuários na BBS.

A BBS também vai promover periodicamente cursos e palestras para seus assinantes. A BBS fica na Avenida Faria Lima, 1390, 8º andar, Grupo 82.

DESTACADORA DE FORMULÁRIOS 100% NACIONAL

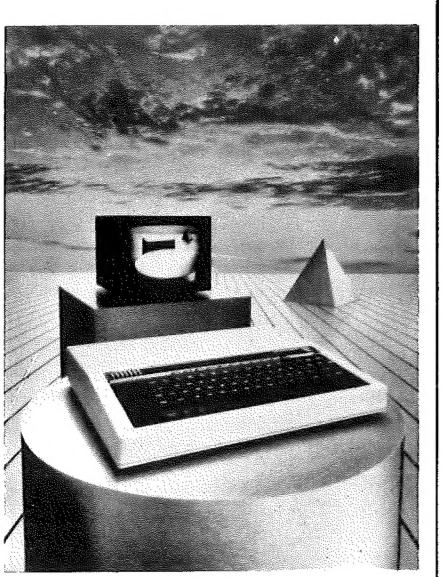
A destacadora SIS 620 da SISTEMATIC foi especialmente desenvolvida no Brasil visando adequar as exigências da contínua evolução do mercado brasileiro de Informática ao momento presente. É destinada aos pequenos Centros de Processamento de Dados e a aplicações onde exista a necessidade de destacar um formulário do outro, sem cortar as remalinas laterais depois de impressos pelo computador. O baixo custo foi possível em virtude das peças e componentes serem 100% nacionais.

O PROJETO DA BBC PARA INICIAÇÃO AOS CONHECIMENTOS DE COMPUTAÇÃO

O Projeto da BBC para Iniciação aos Conhecimentos de Computação foi iniciado em janeiro de 1982, com o objetivo de proporcionar a qualquer pessoa interessada em computação a oportunidade de obter conhecimentos sobre o assunto. Este projeto inclui duas séries para televisão de 10 partes cada, vários livros, cursos por correspondência e *software*, e um sistema de referência e informação através do qual os interessados podem entrar em contato com 1000 fontes diferentes de informação, e com o sistema de microcomputadores da BBC.

O projeto no Reino Unido tem alcançado resultados extremamente positivos: mais de 8 milhões de pessoas assistiram a pelo menos um dos programas de televisão, mais de 200.000 pessoas utilizaram o serviço de referência, 200.000 livros e US$ 1 milhão de *software* foram vendidos, e a ACORN vendeu mais de 170.000 microcomputadores BBC. Os programas de televisão e outros materiais têm sido muito usados em vários países.

CONTATO: John Radcliffe
Room 913, Villiers House
Ealing, London W 5



O ENDEREÇO DE TODOS OS MICROS

Em nossa loja somos todos Pró-informática, Pró-didática e Pró-eletrônica.

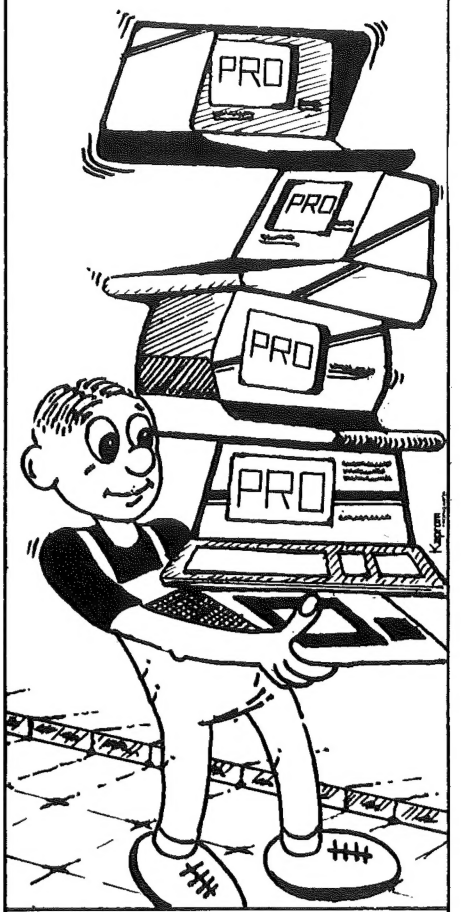
Sysdata ZIROK FLEXIDISK

MICRODIGITAL DACTARI

Polymax Unitron

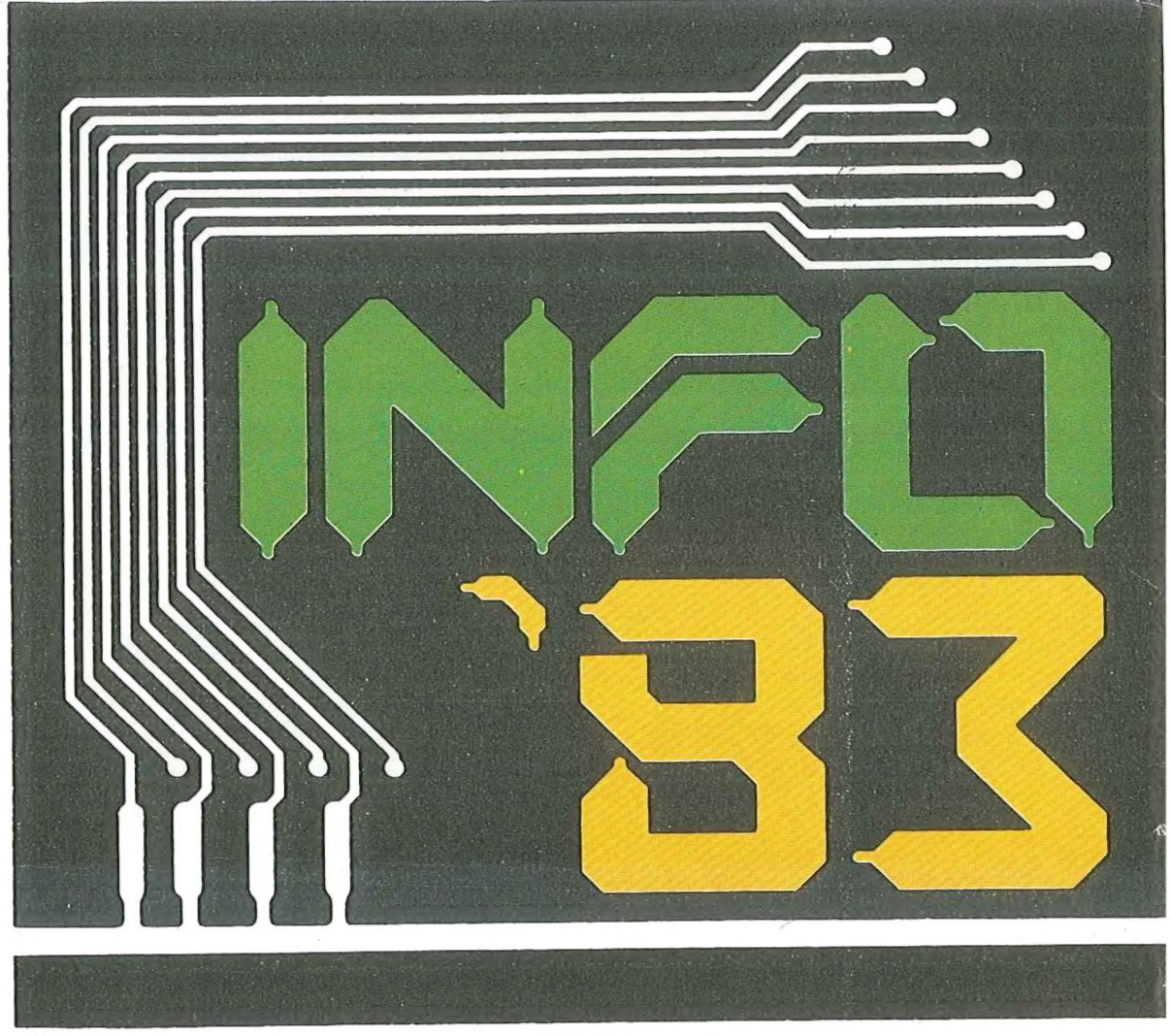
ELEBRA ACECO

PROLOGICA microcomputadores apple



PRÓ ELETRÔNICA
PRÓ ELETRÔNICA COMERCIAL LTDA.
Rua Santa Efigênia, 568 – CEP 01207 – São Paulo – SP
Tels.: 220-7888 – 221-9055 – Telex (011) 34901 – POEC

QUANDO A FANTASIA SE TORNA REALIDADE



Bárbara Hartz

"Ainda estamos embriagados com a informática, mas não vamos ficar de porre". A observação de José Roberto Faria Lima, presidente da III Feira Internacional de Informática, que se realizou em paralelo com o XVI Congresso Nacional de Informática, no Parque Anhembi, em São Paulo, de 17 a 23 de outubro, expressa bem a sensação demonstrada pelas 300 mil pessoas que, de uma forma ou de outra, participaram desses eventos. Em ambos, que constituíram a Informática/83, patrocinada oficialmente pela SEI — Secretaria Especial de Informática e promovida pela SUCESU NACIONAL — Sociedade dos Usuários de Computadores e Equipamentos Subsidiários, se verificou a vitalidade de um setor da economia brasileira que, ao contrário dos outros, cresce da ordem de 30% ao ano, com estimativa de crescimento de 40% para este ano. Tanto em um quanto em outro pode-se constatar a maturidade desse novo empresariado nacional, que constituiu mais de 70% dos 300 e poucos expositores e 270 stands instalados nos 22.000 m^2 da Feira e se manifestou clara e categoricamente pela manutenção da reserva de mercado da área, revelando estar disposto a não perder o que conquistou.

"É gratificante constatar que o Brasil vem perseguindo com tenacidade a meta da capacitação tecnológica no tratamento automático da informação e, para tanto, desvendou seus próprios caminhos", disse Joubert de Oliveira Brízida, Secretário da SEI, órgão do Ministério das Comunicações, em seus discurso de abertura da Feira. Afirmou ainda, na presença do ministro Extraordinário para Assuntos Fundiários e secretário-geral do Conselho de Segurança Nacional, Gen. Danilo Venturini, entre outras autoridades presentes na ocasião, que essa "palpável realidade" da indústria da informática no País tem como missão a desempenhar, concorrer com o seu desenvolvimento para a solução da difícil crise econômica em que estamos engolfados".

Finalizando seu discurso, Brízida chamou a atenção para o que consiste na "grande conquista da comunidade de informática: o reconhecimento da sociedade brasileira de que o importante não é somente o estabelecimento de um parque computacional em nosso país, mas que este venha acompanhado da legítima capacitação científica e tecnológica, que nos permita decidir com autonomia sobre o nosso futuro".

Curiosos e Aprendizes na Terceira Maior Feira do Mundo

Movidos pelo *slogan* "Informática a Serviço da Sociedade: Presente e Futuro, nós procuramos sair do alcance unicamente dos técnicos", explicava na ocasião Salvador Perroti, Presidente de Informática/83, "para discutir temas de impacto à toda a sociedade, tentando conscientizar vários de seus segmentos sobre o que é a computação e como utilizá-la".

De fato, tudo indica que a Feira e o Congresso se conduziram homogeneamente, transformando essa preocupação em realidade. Sob as luzes, cores e sons emitidos pelos stands sempre lotados, as mais diferentes pessoas tiveram a oportunidade de se identificar com seus personagens nos filmes de ficção, dedilhando pela primeira vez o teclado de um computador ou de um microcomputador — a grande sensação da feira — orientados pelos técnicos de plantão. Em particular, um público muito grande e muito jovem, que variava entre 10 e 16 anos, aguçou a sua curiosidade quanto às diversas utilizações dos equipamentos expostos, e não deixou também, por outro lado, de se divertir com a ampla gama de videogames presentes. No circo montado no pátio do Anhembi, as crianças aprendiam a mexer nos computadores sob os olhares atentos dos psicólogos e pedagogos, especialmente contratados para observar as reações causadas por esse estudo através das máquinas.

Avaliando que a Feira cumpriu um de seus grandes objetivos, o institucional, Faria Lima disse ter sido esta uma feira "diferente das outras no mundo". Segundo ele, a pretensão era a de "fazer com que o grande público brasileiro, que ainda não estava tão desperto para a nova realidade da sociedade informatizada, tomasse contato com essa nova era da informação, as suas implicações sociais e políticas e os instrumentos comuns no seu dia-a-dia — computadores, micros, e toda essa aparelhagem que está sendo mostrada aqui".

Entusiasmado com o sucesso da III Feira Internacional da Informática, Faria Lima citou outros dois objetivos a que ela se propunha — o político e o mercadológico — considerando-os atingidos. Conforme informou, o sentido político ocorreu na medida em que todas as entidades liga-

das à área — SUCESU, APPD, ANDEI, ABICOMP, etc — se utilizaram da Feira "como um palco de um forum privilegiado de discussões sobre os temas que as afligem".

Definindo a Feira como "um evento mercadológico", seu Presidente observou que "as empresas brasileiras já tem maturidade, qualidade e disposição de ingressar no mercado internacional", mas que, no entanto, "o mercado brasileiro ainda é muito grande para ser desbravado". Continuando, Lima disse que "deveriam ocorrer modificações no plano do ensino, para introduzir o processamento de dados no primeiro ciclo e todas as formações profissionais de nível superior e secundário conterem noções de informática para cada especificidade — do engenheiro ao advogado, dentista e jornalista".

Observou ainda o crescimento da Feira em relação a 1981, quando haviam somente 141 expositores dispostos em 12 mil m^2. "Sem dúvida nenhuma, o setor de micros atraiu grande parte da atenção do público, seguido pelas empresas nacionais e sua nova geração de produtos. Sentimos também uma diferença qualitativa em relação às feiras anteriores; com a presença de inúmeras *software houses*. Além disso, a participação da área de controle de processos, de comunicações, os setores de equipamentos auxiliares, de móveis, constituem todo um setor produtivo voltado para essa nova realidade, que está sendo apresentada no Brasil de forma acelerada, principalmente nos principais centros, como é o caso de São Paulo".

Congresso tira as cartas do colete

No Congresso, desdobrado em várias atividades, desde as plenárias, onde predominou a pauta política, até às

sessões especiais, com temas tão distintos quanto Telemática/83, Inteligência Artificial e Software Linguagem de Alto Nível, notou-se a presença significativa de pessoas que não lidam no seu cotidiano com a informática. Também nos cursos ministrados durante o Congresso isso ocorreu. Cerca de 160 alunos freqüentaram os 7 Mini-Cursos, dedicados a diversas categorias profissionais — advogados, arquitetos, médicos e dentistas, executivos e administradores — e mais de 250 fizeram o Curso de Introdução ao Microcomputador. Mario Rosenthal, diretor de Programação de Cursos do Congresso, disse acreditar que "os cursos permitiram o primeiro contato estruturado desse pessoal com a informática".

Atendendo ao objetivo de popularizar a informática, se inscreveram também 2.400 estudantes nas sessões dedicadas a eles. Porém, além dos leigos, os especialistas tiveram a oportunidade de trocar toda e qualquer experiência no Congresso — da técnica à política. Tentando desvendar talentos, foram eleitos 27 dos mais de 200 especialistas que enviaram trabalhos, apresentados nas Sessões Técnicas e mostrando novas soluções na área de processamento em tempo real e teleprocessamento.

Destacou-se no Congresso a presença internacional com o comparecimento das 24 personalidades convidadas — que incluiam latino-americanos, franceses, americanos e japoneses para expor os aspectos mais teóricos e novas tecnologias nas Sessões Especiais. A maioria dos latino-americanos como são membros da Federação Latino-Americana de Usuários de Informática — FLAI, participaram das reuniões ordinárias dessa entidade, que também ocorreram nesses dias.

O cenário do usuário nacional foi constituído, principalmente, pelos foruns de debates. Lá eles se posicionaram a respeito de quatro temas: reiteraram o apoio à regulamentação da profissão de analista de sistemas e todas as outras referentes ao processamento de dados, através de encaminhamento de substitutivo ao projeto que se encontra no Senado e à formação do sindicato da categoria desses profissionais; denunciaram por meio de documento da SUCESU NACIONAL a interferência do estado das diversas áreas de atuação da informática, entre as quais a fabricação de equipamentos, "que são fornecidos de forma monopolítica às instituições governamentais"; e fizeram a avaliação dos modelos industriais brasileiros e da política de software.

Salvador Perroti, que também é o atual presidente da FLAI, observou que, "pela primeira vez o Congresso de Informática abriu o forum de debates para discussão desse assunto às demais entidades como a APPD, ABICOMP, ABINEE, ASSESPRO e SBC, entre outras.

RESERVA DE MERCADO

A grande vedete nas discussões políticas

De início, a discussão em torno desse assunto se apresentou com certa apreensão. Na primeira e uma das mais importantes plenárias, a de Política de Informática, o Cel. Edison Dytz, Secretário Executivo da SEI, disse que a política de reserva de mercado poderia sofrer mudanças com o fim do Governo do Presidente Figueiredo, se até o Congresso Nacional não aprovasse projeto de lei transformando em lei a política imposta pela sua secretaria. Seguia o mesmo temor já manifestado anteriormente por Salvador Perroti, em ocasiões anteriores. Apenas o Gen. Venturini destoava um pouco, afirmando nunca ter sofrido pressões para a quebra da reserva de mercado.

Posteriormente, após os diversos pronunciamentos ao tema, o tom dos discursos já adquiria mais confiança. Naturalmente, isso ocorria depois do comprometimento de vários políticos em levar adiante a discussão dentro do Congresso Nacional. O Senador pedessista Carlos Chiarelli, após longa história da CAPRE — Comissão Coordenadora das Atividades de Processamento Eletrônico contada nessa mesma plenária por Ricardo Saur Ex-Secretário Executivo desse órgão, relatando todas as dificuldades encontradas pa-

ra que se chegasse a essa política dentro da SEI, disse que "a democracia existente na atual política de informática é quem garantirá essa reserva de mercado". Por outro lado, o Senador Marco Maciel, embora defendendo que a transformação em lei da reserva de mercado deva partir de iniciativa do Executivo, comentou que "os segmentos dos minis e posteriormente o dos micros e seus periféricos foram, acertadamente, definidos como espaços destinados à iniciativa privada nacional, criando-se desse modo as condições objetivas para o nascimento e posterior consolidação da indústria nacional de informática, cujos resultados são incontestáveis".

Perroti, nos dias que se seguiram à abertura do Informática/83, dizia achar que "já há um consenso geral da sociedade sobre a reserva de mercado". Observou ele ter sido "muito importante a presença dos políticos, pois o momento requer que se faça a institucionalização legal das medidas que venham consolidar o nosso setor, o que só será conquistado com a participação consciente dos políticos e do empresariado nacional".

Evidentemente, mais do que avançar na discussão e demarcar a importância da reserva de mercado fazer parte da legislação brasileira não seria possível, em reuniões deste tipo. O fato é que a discussão se alastrou pela sociedade a partir do Congresso de Informática, seja pelos pronunciamentos dos mais significativos representantes da indústria de informática, seja pela repercussão na imprensa. Até que isso se efetive, um longo e controvertido caminho ainda terá que ser percorrido. O próprio Senador Marco Maciel admitiu, naqueles dias, ser "natural que haja interesses estrangeiros nessa área", dizendo que "isso não quer dizer que a vontade nacional deva se curvar pura e simplesmente perante eles". A pressão do capital estrangeiro para aprofundar a sua participação nesse setor de nossa indústria se evidenciou em reunião de vários empresários de diferentes ramos com o referido Senador para lhe dar subsídios ao envio de projeto de lei para o Congresso Nacional, quando um executivo da IBM se manifestou por uma legislação aberta, que não restrinja a origem do capital.

O Ministro da Indústria e Comércio, Camilo Penna, ao defender a plena robotização da indústria exportadora brasileira na sessão Informática e Emprego, se mostrou favorável à formação de *joint-ventures* nesse setor econômico, quando Brízida, por outro lado, considerou durante o Congresso "não ser conveniente" a formação de *joint-ventures* de capital associado à tecnologia.

Destaque: proteção à tecnologia e ao software nacional

Outro tema político de relevância foi levantado na solenidade de abertura do XVI Congresso, pelo presidente da SUCESU NACIONAL, José Henrique Portugal, que chamou a atenção para a "urgência da criação de uma lei que proteja e defina o software, permitindo a sua inclusão como garantia real de financiamento para novo software e que o transforme em lastro de um fundo seguro, como solução alternativa".

Comparando a atual situação da indústria de informática com a da automobilística em 1950, argumentou a favor da "inteligência nacional" para que as empresas e empresários brasileiros possam "andar por suas próprias pernas comandados pelos seus próprios cérebros, sem as caixas pretas de tecnologias não dominadas". Nesse sentido, falou também que os usuários, aos quais representa, não desejam que o Brasil seja apanhado pela crise do "chip", do "software", do "protocolo", "ou de qualquer nome de crise que encubra uma dominação tecnológica". Para ele, a

LITEC
Livraria Editora Técnica Ltda.

A maior livraria da América Latina especializada em INFORMÁTICA, COMPUTAÇÃO E ELETRÔNICA.
Mais de 3.000 títulos em português, espanhol e inglês em permanente exposição.

Rua dos Timbiras 257 - 01208 São Paulo Tel. (011) 220-8983 cx. postal 30.869

TRANSITRON

Comércio de Componentes

- TTL
- C.MOS
- ICL 7107
- 2114
- 2708
- 2716
- 2732
- LINHA Z80
- LINHA Z80A
- ELETROLÍTICO
- TRANSÍSTOR
- POLIESTER
- TÂNTALO
- PLATE
- RESISTOR
- FUSÍVEL
- SOQUETE
- CONECTOR C. IMP.
- APAGADOR DE EPROM Cr$ 55.000,00

TRANSITRON ELETRÔNICA LTDA.
Rua dos Gusmões, 353, 3º andar, cj. 31
Fones 221-2959 - 221-2701 - 223-5187
Telex - 37982 Representante em B. H.
Rua Eng. Antonio Guerra, 174, cj. 401
Fone 332-9580 - Sr. Rogério.

Um micro só é grande quando tem Assistência Técnica BCD.

Com a Assistência Técnica BCD, você conta com a garantia de receber um serviço recomendado pela Shell, BNH, SERPRO e SENAC. A BCD Engenharia oferece também os melhores Contratos Anuais de Assistência Técnica, que garantem o máximo ao seu micro.

E sem custar mais por isto.

Os micros da Prológica, Spectrum, Digitus, Dismac e muitos outros têm na BCD uma Assistência aprovada pelos próprios fabricantes. Além disso, atendemos também os micros importados Apple, TRS 80 e Sinclair.

Para maior desempenho do seu equipamento, guarde este nome na memória: BCD.

bcd Vida longa para seu micro.

Rua Barão de Mesquita, 663 ljs. 3 e 4 - Tijuca - Tel.: 238-2186

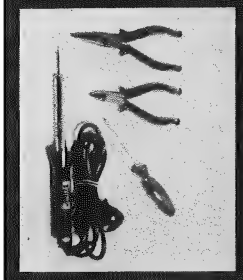
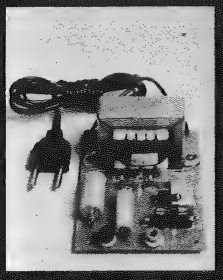
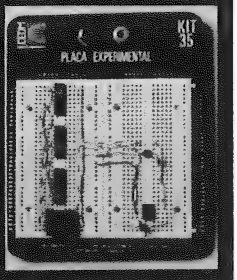
CEDM
CURSOS DE APERFEIÇOAMENTO

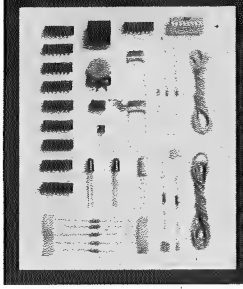
MAIS SUCESSO PARA VOCÊ!

**Comece uma nova fase na sua vida profissional.
Os CURSOS CEDM levam até você o mais moderno ensino técnico programado e desenvolvido no País.**

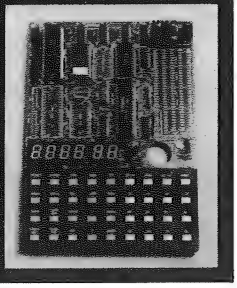
CURSO DE ELETRÔNICA DIGITAL E MICROPROCESSADORES

São mais de 140 apostilas com informações completas e sempre atualizadas. Tudo sobre os mais revolucionário CHIPS. E você recebe, além de uma sólida formação teórica, KITS elaborados para o seu desenvolvimento prático. Garanta agora o seu futuro.

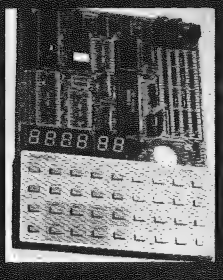
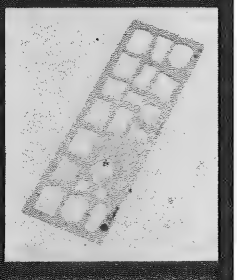


CEDM-20 - KIT de Ferramentas.
CEDM-78 - KIT Fonte de Alimentação 5v/1A. CEDM-35 KIT Placa Experimental
CEDM-74 - KIT de Componentes.
CEDM-80 MICROCOMPUTADOR Z80 ASSEMBLER.



CURSO DE PROGRAMAÇÃO EM BASIC

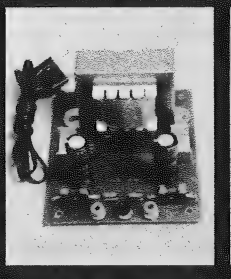
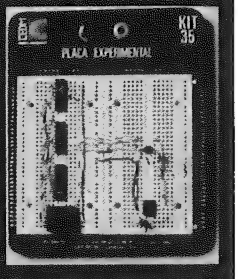
Este CURSO, especialmente programado, oferece os fundamentos de Linguagem de Programação que domina o universo dos microcomputadores. Dinâmico e abrangente, ensina desde o BASIC básico até o BASIC mais avançado, incluíndo noções básicas sobre Manipulação de Arquivos, Técnicas de Programação, Sistemas de Processamento de Dados, Teleprocessamento, Multiprogramação e Técnicas em Linguagem de Máquina, que proporcionam um grande conhecimento em toda a área de Processamento de Dados.

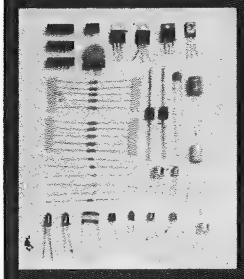
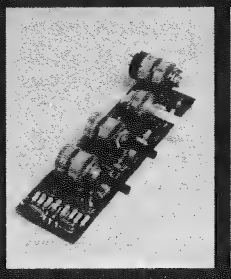
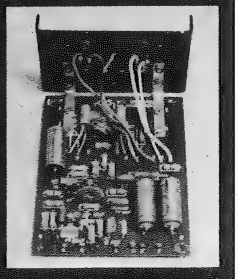



KIT CEDM Z80 BASIC Cientifico.
KIT CEDM Z80 BASIC Simples.
Gabarito de Fluxograma E-4. KIT CEDM SOFTWARE
Fitas Cassete com Programas.

CURSO DE ELETRÔNICA E ÁUDIO

Métodos novos e inéditos de ensino garantem um aprendizado prático muito melhor. Em cada nova lição, apostilas ilustradas ensinam tudo sobre Amplificadores, Caixas Acústicas, Equalizadores, Toca-discos, Sintonizadores AM/FM, Gravadores e Toca-Fitas, Cápsulas e Fonocaptadores, Microfones, Sonorização, Instrumentação de Medidas em Áudio, Técnicas de Gravação e também de Reparação em Áudio.

CEDM-1 - KIT de Ferramentas. CEDM-2 - KIT Fonte de Alimentação + 15-15/1A. CEDM-3 - KIT Placa Experimental
CEDM-4 - KIT de Componentes. CEDM-5 - KIT Pré-amplificador Estéreo. CEDM-6 - KIT Amplificador Estéreo 40w.

Você mesmo pode desenvolver um ritmo próprio de estudo. A linguagem simplificada dos CURSOS CEDM permite aprendizado fácil. E para esclarecer qualquer dúvida, o CEDM coloca à sua disposição uma equipe de professores sempre muito bem acessorada. Além disso, você recebe KITS preparados para os seus exercícios práticos.

Ágil, moderno e perfeitamente adequado à nossa realidade, os CURSOS CEDM por correspondência garantem condições ideais para o seu aperfeiçoamento profissional.

GRÁTIS

Você também pode ganhar um MICROCOMPUTADOR.

Telefone (0432) 23-9674 ou coloque hoje mesmo no Correio o cupom CEDM.

Em poucos dias você recebe nossos catálogos de apresentação.

CEDM

**Avenida São Paulo, 718 - Fone (0432) 23-9674.
CAIXA POSTAL 1642 - CEP 86100 - Londrina - PR**

CURSO DE APERFEIÇOAMENTO POR CORRESPONDÊNCIA

Solicito o mais rápido possível informações sem compromisso sobre o CURSO de . . . . . .

Nome. . . . . .

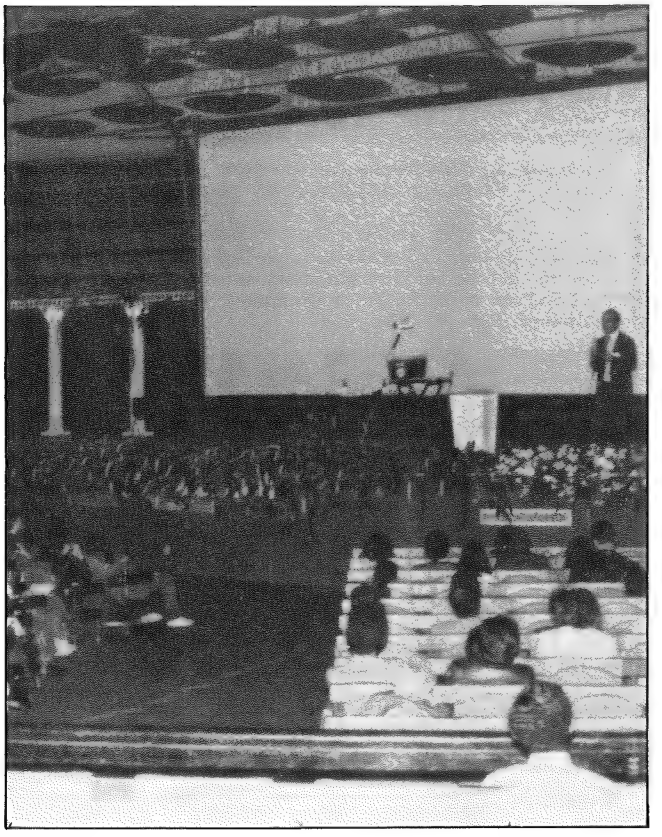
Rua. . . . . .

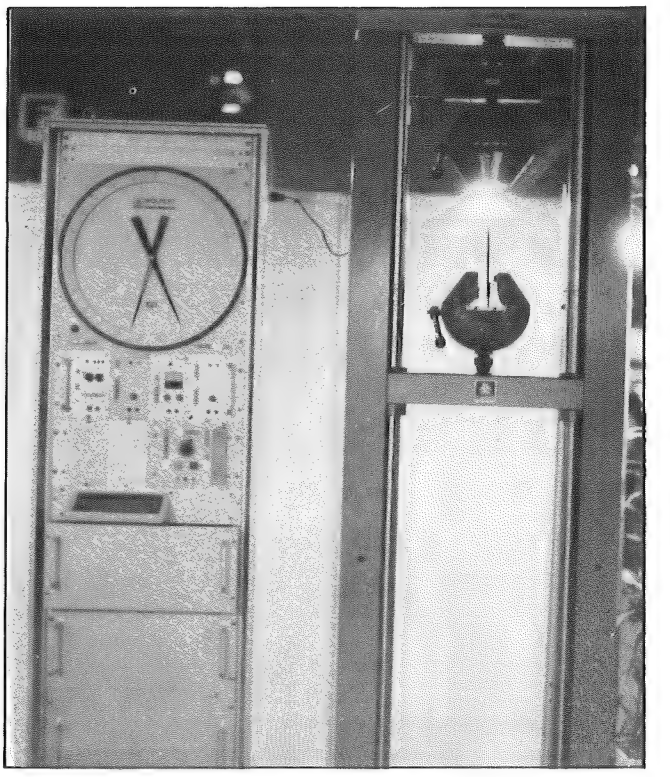
Cidade . . . . . .

Bairro. . . . . CEP . . . . . .

questão fundamental que interessa ao País é "a capacitação tecnológica em bases reais — auxiliada pela reserva de mercado — com profunda consciência de suas implicações, cabendo ao consumidor a determinação da direção do mercado". Salientou ainda a importância da "participação das instituições de ensino e pesquisa nesse processo", abordando outros pontos básicos: a necessidade do debate das condições de produção das novas tecnologias no País; a imperiosidade do "acompanhamento das práticas comerciais pelo estado, pelo governo e principalmente pelas entidades de classe do setor"; a facilidade de se chegar à discussão de novas tecnologias adequadas ao País através da compatibilidade de processos, equipamentos, pacotes, protocolos e afins. Nesse último aspecto, Portugal ressaltou a importância da discussão em torno dos "superminis", "supermicros" e "micrões", dizendo que "não interessa ao País a compra de nova tecnologia para desenvolver esses aparelhos", porém, se houver compatibilidade a nova tecnologia poderá ser desenvolvida aqui".

Falando-se na tecnologia nacional de informática, novamente a Feira acompanhou as preocupações dos trabalhos que transcorreram no Congresso. A Financiadora de Estudos e Projetos — FINEP, que inclui entre as suas prioridades os setores considerados estratégicos, como a área de microeletrônica e informática, esteve presente à Feira coordenando as atividades de 17 stands, com seus inventos e expositores, que incluem universidades, centros e fundações de Pesquisa e faculdades de engenharia. Carlos Gunen, um dos representantes da FINEP na feira, disse ser este "o terceiro ano consecutivo em que a FINEP patrocina a quase totalidade da área de pesquisa e desenvolvimento da Feira Internacional". Observou também que, durante esses anos, houve importante crescimento do número de participantes dessas instituições que "vêm mostrar a alternativa tecnológica nacional tupiniquim", apresentando em outubro passado "trabalhos de software e hardware, objeto de seus estudos no último ano".





Em defesa da cooperação latino-americana

Inúmeros foram os pronunciamentos dos pesquisadores estrangeiros e representantes de sociedades de usuários de informática de países da América Latina, para que estes se unam formando uma rede de cooperação nessa área. Renato Mazola, Assessor da Presidência da SUCESU NACIONAL e diretor da FLAI, explicou porque é lógica esta posição. Pergunta ele: "Se nós falamos que a informática não tem fronteiras, por que não criamos a integração entre os países latino-americanos"? Respondendo em seguida que "falamos a mesma linguagem, temos os mesmos problemas — falta de divisas, falta de recursos nas universidades — então, se nos pautarmos pelo modelo dos países desenvolvidos, estabeleceremos uma defasagem muito grande".

Para facilitar essa cooperação latino-americana o II Encontro Latino-Americano dos Usuários de Informática decidiu propor a criação de um Dia da Informática, a ser comemorado em todos os países da América do Sul. Essa data será estabelecida em abril do próximo ano, na Exposuária da Argentina. No encontro, foi reiterada ainda a necessidade de tornar a FLAI a única entidade representativa dos usuários latino-americanos, para que não haja nenhum tipo de confusão com a CALAI — Centro Latino-americano de Informações, que é um órgão governamental. Compareceram ao encontro as seguintes entidades sul-americanas: APUD — Associación Paraguaia de los Usuários de Informática, representada por Cibar Cáceres; ACUC — Associación Colombiana de los Usuários de Computadores, por José Jaime Orduz; APCI — Associación Peruana de Computos e Informática, Diomedes Sarmiento; SUI — Sociedade Uruguaia de Informática, Horacio Madorno; e USUARIA — Sociedade de Usuários de los Computadores, Artonio Castro Lechtaler.

Compareceram também ao Congresso outros técnicos de vários países, entre eles Estados Unidos, França, países da América do Sul, Senegal, Costa do Marfim e Marrocos, além de 50 estudantes uruguaios de computação e alguns outros argentinos.

consultoria

Trata-se de uma coluna mensal destinada a formar um elo entre o usuário e o fabricante de equipamentos abrindo-se ao leitor para recebimento de cartas que contenham perguntas sobre funcionamento, operação, preços, modificações e falhas em microcomputadores e periféricos de fabricação nacional ou importados.
Se você tem dúvidas sobre algum assunto ligado à informática, escreva para:
CONSULTORIA – REVISTA INTERFACE – ESTRADA DO TINDIBA, 2380 – JACAREPAGUÁ – RJ – CEP 22700.

P Possuo um microcomputador TK85 e gostaria de saber os endereços e funções dos comandos DSAVE, DVERIFY, DLOAD, DHSAVE e DHLOAD.

R O comando *DSAVE* é responsável pela gravação do arquivo e seu endereço inicial é 8288. O *DVERIFY* verifica a gravação e seu endereço inicial é 9816. O *DLOAD* lê a fita e armazena no *buffer* indicado; seu endereço inicial é 8305. O *DHSAVE* grava na fita em *Hi-speed* e seu endereço inicial é 9008. O *DHLOAD* lê a fita em *Hi-speed* e seu endereço inicial é 9189.

P Como as fitas cassetes estão sujeitas a desmagnetizações acidentais ou desgastes, como fazer para salvar os meus programas automaticamente, sem ter que transferi-los para novas fitas?

R A rotina a seguir salva o seu programa de modo automático; acrescente-a aos seus programas. Para salvá-los digite RUN 8000 e responda ao INPUT da linha 8000 com o nome do programa. É fundamental que seu micro seja compatível com o SINCLAIR.

```
8000 INPUT X$
8005 LET X = PEEK 16388
8010 POKE 16388,3
8020 POKE 16388, PEEK 16388 -1
8030 SAVE X$
8040 IF PEEK 16388 THEN GOTO
     8020
8045 POKE 16388, X
8050 RUN
```

P Qual a diferença entre as memórias bipolares e CMOS?

R Existem duas tecnologias de construção das memórias: com transistores bipolares ou com transistores CMOS. A principal diferença entre elas é no tempo de acesso, ou seja, no intervalo entre o aparecimento do endereço e o surgimento da informação na saída. Nas memórias bipolares é da ordem de 50 a 90 nanosegundos, enquanto nas memórias CMOS é bem maior.

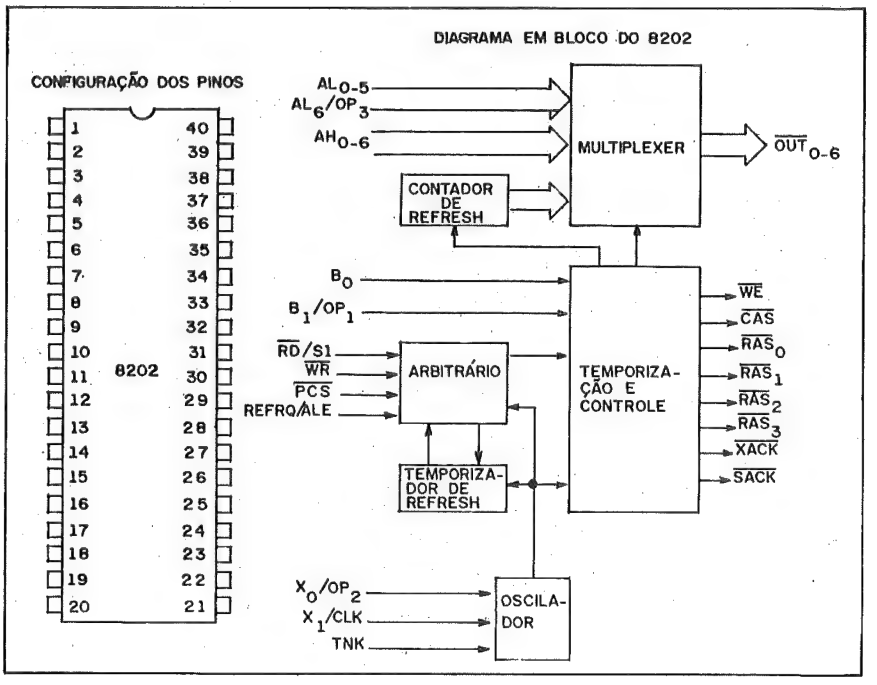
As memórias bipolares são disponíveis com capacidade de 1Kb, 2Kb e 4Kb, enquanto as memórias CMOS são disponíveis em até 16Kb. As memórias EPROM's só são construídas com tecnologia CMOS, atingindo tempos de acesso na ordem de 5 a 10 nanosegundos.

P Gostaria de saber como funciona o controlador de memória 8202 da Intel e se o mesmo já faz o *refresh* das memórias RAM's ou são necessários outros circuitos auxiliares.

R O 8202, controlador dinâmico de RAM, foi desenvolvido para fornecer todos os sinais necessários para as memórias RAM 2104A, 2116 e 2117 em sistemas de microcomputadores. Apresenta as seguintes facilidades:

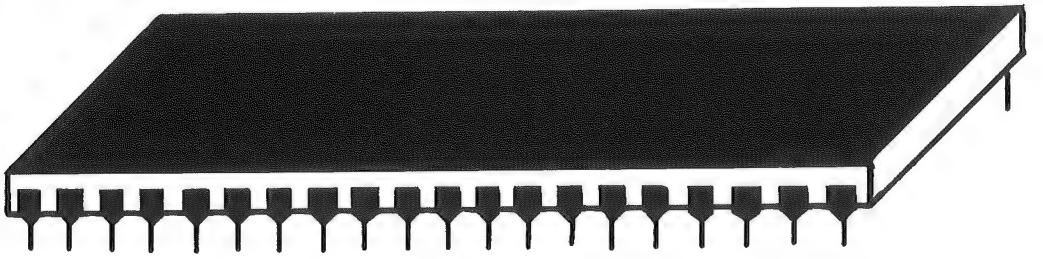
- Sinais para controle das memórias RAM's;
- Endereçamento direto e drives para 64Kb de memória, sem drivers externos;
- Endereçamento multiplexado e strobes;
- Temporizador de *refresh* e contador de *refresh;*
- Ciclos de *refresh* internos ou externos;
- Totalmente compatível com os microprocessadores 8080A e 8085A.

A pinagem e o diagrama em bloco é apresentado a seguir.



68 000

O PODEROSO MICROPROCESSADOR DE 16 BITS



1ª Parte

Por Cesar da Costa

Os microcomputadores de 16 bits são a última novidade na prancheta dos projetistas de micros nacionais. A maioria desses projetistas não teve tempo suficiente para escolher qual o melhor microprocessador disponível no mercado (apesar da INTERFACE número 2 e 3 apresentar uma comparação entre os três principais). Alguns optaram pela família Intel (8086) pois estavam acostumados a trabalhar com os seus antecessores de 8 bits (8080/8085) e outros, pelo mesmo motivo, optaram pelo Z8000 da Zilog.

Após algum tempo de pesquisa e observação prática em outros campos de aplicação tais como processamento de sinais digitais e computação gráfica de alta resolução (terminais gráficos), onde houve necessidade de substituição dos microprocessadores *bit-sliced* de alto custo por microprocessadores de 16 bits de igual desempenho e menor custo, concluímos em optar pelo melhor e mais poderoso microprocessador disponível no mercado: o 68000.

Neste artigo apresentamos suas principais características, que farão com que muitos projetistas reflitam sobre a sua escolha pelo microprocessador de 16 bits ou reformulem seus projetos em função desse microprocessador. Alertamos aos que dele venham a se utilizar para a importância de não se restringirem aos limites deste artigo, uma vez que jamais poderá cobrir em extensão e profundidade o assunto.

Em 1972 era lançado pela Intel Corporation o primeiro microprocessador: O *4004.* Tratava-se de um microprocessador de 4 bits, que tinha o objetivo de reduzir os custos de hardware, substituindo algumas centenas de chips de lógica TTL convencional. Daquele humilde início os microprocessadores evoluíram em tamanho de palavra, tamanho de barramento e potência de processamento. Primeiro foram desenvolvidos os microprocessadores de 8 bits, exemplificados pelas famílias 8080, 8085, Z-80, 6800, 6502 etc. Depois foram os de 16 bits, e a última palavra são os microprocessadores de 32 bits, que logo farão sua estréia.

Com a evolução da potência de processamento, as aplicações começaram a mudar. Os microprocessadores de 8 bits tornaram-se poderosos o suficiente para suportar linguagens de alto nível como BASIC, PASCAL e outras, tornando os computadores pessoais uma realidade. O novo cotidiano dos microprocessadores de 16 bits é tão forte que está proliferando outros novos produtos.

Os principais microprocessadores de 16 bits disponíveis no mercado são o Intel 8086, o Zilog Z8000 e o MC 68000 (vide a comparação entre estes microprocessadores na INTERFACE número 2 e 3). O MC 68000 é fornecido por diversos fabricantes, incluindo Motorola, Mostek e Signetics. Estes microprocessadores estão sendo utilizados com excelentes resultados em aplicações especiais tais como processamento de sinais digitais e computação gráfica de alta resolução. Estas áreas, não fazem muito tempo, eram esfera de ação dos dispendiosos microprocessadores *bit-sliced.*

ARQUITETURA DO MC 68000

Para avaliar a potência do MC 68000 e como ele vai de encontro às necessidades de suas novas aplicações, temos que analisar sua arquitetura: registros, conjunto de instruções e outras capacidades.

O MC 68000 é uma unidade de processamento de 16

bits, sendo que seu *barramento de dados* tem capacidade para palavras de 16 bits. Contudo, todos os seus registros internos têm capacidade para palavras de 32 bits. Por isso, algumas vezes é citado como o microprocessador de 16/32 bits. Em comparação, o 8086 e Z8000 usam somente registros internos de 16 bits. O registro de dados do MC 68000 pode manipular palavras de 8, 16 ou 32 bits.

A figura 1 apresenta a organização interna do MC 68000. Existem 18 registros de 32 bits, assim constituídos:

- *8 registros de dados (D0 – D7)*
- *7 registros de endereço (A0 – A6)*
- *1 registro de endereço A7 (constituído de dois ponteiros de pilha – SP).*
- *1 contador de programa (PC).*

O registro de estados (status) é de 16 bits e consiste de dois bytes: um byte para o usuário e outro byte para o sistema (supervisão).

Os dados podem ser manipulados em bits, bytes (8 bits), palavras (16 bits) e palavras extensas (32 bits).

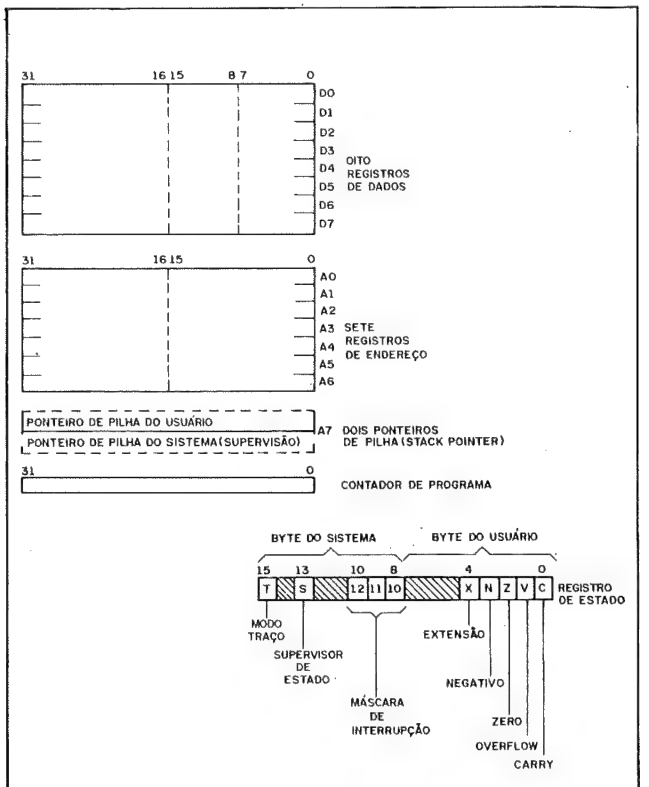


FIGURA 1 – ORGANIZAÇÃO INTERNA DO MC 68000

A figura 2 apresenta a Organização de memória utilizada pelo MC 68000, que pode endereçar diretamente 2^{24} (16777216) bytes, arranjados em seqüência linear. Os endereços são sempre bytes de endereço, isto é, palavras de 16 bits são localizadas por bytes de endereço par e podem somente ser referenciadas por aquele endereço. Palavras extensas de 32 bits são localizadas por palavras de endereço. Um byte pode ser endereçado em uma localização par ou ímpar e a metade superior ou inferior da palavra é utilizada. A palavra é obtida em um acesso, embora leve dois acessos para buscar uma palavra extensa: metade superior (mais baixa palavra de endereço) e, então, metade inferior (próxima palavra de endereço). O I/O é mapeado na memória.

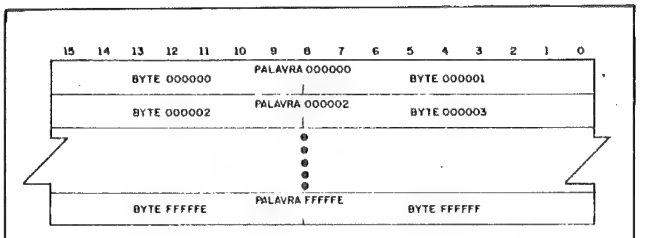


FIGURA 2 – A ORGANIZAÇÃO DE MEMÓRIA PODE SER DIRETAMENTE ENDEREÇADA POR 2^{24} BYTES.

CONJUNTO DE INTRUÇÕES

O conjunto de instruções do MC 68000 é apresentado na tabela 1. Notamos que existe somente 56 *instruções básicas,* sendo que muitas delas têm variações. O conjunto de instruções foi desenvolvido dentro da filosofia de que um pequeno número de instruções é mais fácil de ser lembrado e manipulado pelo programador do que um imenso repertório de instruções, desde que este pequeno grupo de instruções seja poderoso.

As instruções usuais são encontradas no conjunto: aritméticas (incluindo multiplicação e divisão), lógicas (AND, OR, EOR etc), desvios, deslocamentos, rotações e comparações. A instrução de movimento de dados é o *MOVE*, uma das mais poderosas do conjunto. Esta instrução pode tomar qualquer modo de endereço efetivo para um ou outro operando e pode transferir dados de algum lugar para outro (memória para memória, memória para registro etc). A potência da instrução *MOVE* será apresentada posteriormente, quando analisarmos os modos de endereçamento do MC 68000.

Em adição, as instruções básicas, a tabela 1 apresenta algumas instruções de propósito especial, projetadas especificamente para suportar os geradores de código para linguagens de alto nível tal como o PASCAL. São elas:

- *Instrução* LINK *e* UNLK *(unlink)*
- *Instrução* LEA *(Load effective address) – endereço efetivo de carga*
- *Instrução* PEA *(Push effective address) – endereço efetivo de* push
- *Instrução* CHK *(Check de registros)*

A instrução *LINK* implementa o procedimento de chamada da linguagem de alto nível. Na figura 3 (A) a antiga forma do ponteiro de pilha (FP) igual a A_2 é salva por um avanço na pilha. A nova forma do ponteiro é criada pela colocação do ponteiro de pilha (SP) em A_2, figura 3B, e, finalmente, variáveis locais são alocadas na pilha, figura 3C,

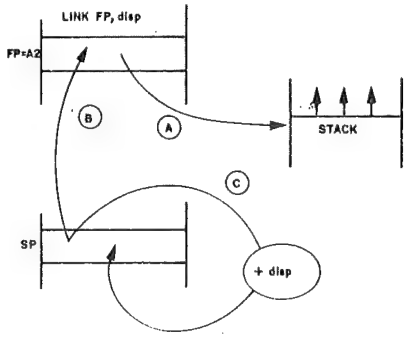


FIGURA 3 – COMO A INSTRUÇÃO LINK É IMPLEMENTADA.

por adição de deslocamento no ponteiro de pilhas. Uma instrução de sub-rotina *JUMP* é então usada para o procedimento de entrada.

A instrução *UNLK* reverte este procedimento pelo rearmazenamento do ponteiro de pilha (realocação do espaço das variáveis locais) e atração do ponteiro (FP) anterior. A reentrada de programas é muito mais rápida e suave para implementação com essas instruções.

Das instruções mais úteis, temos a de *decremento* e de *desvio* no código de condição (DBcc), isto é, um laço primitivo que desvia se a condição não é encontrada ou se o contador não decrementa para -1. Então, elas recolocam uma seqüência de cinco instruções: CMP, BRct, SUB, CMP e BRcc. Isto acelera os laços, muito comuns, encontrados em todos os programas.

No próximo número apresentaremos os modos de endereçamento, sistema de proteção e barramentos.

TABELA 1 – RESUMO DO CONJUNTO DE INSTRUÇÕES DO MC 68000

MNEMÔNICO	DESCRIÇÃO
ABCD	*Soma decimal com prolongamento*
ADD	*Soma*
AND	*Lógica And*
ASL	*Deslocamento aritmético à esquerda*
ASR	*Deslocamento aritmético à direita*
Bcc	*Desvio condicional*
BCHG	*Teste bit e mude*
BCLR	*Teste bit e limpe*
BRA	*Desvio contínuo*
BSET	*Teste bit e* set
BTST	*Teste bit*
CHK	*Check registro sem limite*
CLR	*Limpe operando*
CMP	*Comparação*
DBcc	*Condição de teste, decremento e desvio*
DIVS	*Divisão indicada*
DIVU	*Divisão não indicada*
EOR	*Exclusive or*
EXG	*Registro de troca*
EXT	*Sinal de prolongamento*
JMP	*Jump*
JSR	*Jump para sub-rotina*
LEA	*Endereço efetivo de carga*
LINK	*Pilha link*
LSL	*Deslocamento lógico à esquerda*
LSR	*Deslocamento lógico à direita*
MOVE	*Transferência*
MOVEM	*Transferência de registros múltiplos*
MOVEP	*Transferência de dados periféricos*
MULS	*Multiplicação indicada*
MULU	*Multiplicação não indicada*
NBCD	*Negativo decimal com prolongamento*
NEG	*Negativo*
NOP	*Operação not*
NOT	*Complemento de uns*
OR	*Lógica or*
PEA	*Endereçamento efetivo de* push
RESET	*Reset para dispositivos externos*
ROL	*Rotações à esquerda sem prolongamento*
ROR	*Rotações à direita sem prolongamento*
ROXL	*Rotações à esquerda com prolongamento*
ROXR	*Rotações à direita com prolongamento*
RTE	*Volta de desaprovação*
RTR	*Volta e rearmazenamento*
RTS	*Volta de sub-rotina*
SBCD	*Subtração decimal com prolongamento*
Scc	*Condicional* set
STOP	*Parada*
SUB	*Subtração*
SWAP	*Troca metade do registro de dados*
TAS	*Testa e* seta *operando*
TRAP	*Armadilha*
TRAPV	*Armadilha em* overflow
TST	*Teste*
UNLK	*Não link*

ORT

INSTITUTO DE TECNOLOGIA ORT

CURSOS DE PROCESSAMENTO DE DADOS

FORMAÇÃO DE PROGRAMADORES (COMPLETO)
Duração: 8 meses
Horário: 2ª a 5ª feira, de 19:00 às 22:00 hs.

MICROCOMPUTADORES E A LINGUAGEM BASIC
Duração: 3 semanas
Horário: 2ª a 5ª feira, de 19:00 às 22:00 hs
Turmas de 15 alunos

AMPLA UTILIZAÇÃO DO IBM-4341 E DO LABORATÓRIO DE MICROCOMPUTADORES

Visite o CPD-ORT – Diariamente após 13:00 hs – Rua Dona Mariana, 213 – Botafogo – Rio de Janeiro – Tels.: 226-3192 e 246-9423

MICRO.BYTE
SISTEMAS E EQUIPAMENTOS COMPUTACIONAIS LTDA.

HARDWARE
• Periféricos para Linha Sinclair TK-82, TK-85, NEZ8000 e CP-200

SOFTWARE
• Jogos e Aplicativos
• Desenvolvimento de Sistemas Próprios

CURSOS
• Linguagem Basic com aulas práticas
• Apostilas Grátis
• Desenvolvimento de Programas

BUREAUX DE SERVIÇOS
• Administração. • Contabilidade
• Folha de Pagamento • Estoque • Etc.

MERCADO DE MICRO USADOS
• Agenciamento
• Compra e Venda

Rua Buenos Aires, 41 - 3º andar, Centro
CEP: 20.070 — Rio de Janeiro — RJ
Telefone: (021) 263-4024

A. T. Publicidade Ltda.

Maxxi, o micro pessoal muito profissional da Polymax.

ASSISTÊNCIA TÉCNICA EM TODO O BRASIL.

Agora vamos processar os dados técnicos dele.

O **Maxxi** é um microcomputador pessoal — profissional de grande versatilidade e assegurada possibilidade de expansão. **Compatível com APPLE II PLUS*, aceita mais de 5 mil programas aplicativos disponíveis no mercado.** Sua característica padrão inclui um monitor e linguagem Polysoft Basic, ambas gravadas em ROM, com 2 kbytes e 10 kbytes, respectivamente, 48 kbytes de memória RAM disponíveis para o usuário; interface para gravador cassete, vídeo e tv colorida (sistema PAL-M); teclado padrão ASC II e fonte de alimentação, dispostos em um gabinete próprio.

Veja aqui sua essência técnica:

Microprocessador
6502 operando com freqüência de 1 MHZ.

Vídeo
O **Maxxi** possui um vídeo profissional de 12'' com fosfatização verde e pode conectar-se também com uma televisão comum (colorida ou preto & branco), operando no modo **texto** ou **gráfico** (baixa ou alta resolução), sendo completamente transparente ao usuário o acesso à memória. No modo **gráfico,** as últimas 4 linhas do vídeo operam no modo **texto.** Todos os modos de operação com o vídeo são selecionáveis por Software.

Modo Texto
- 40 caracteres/linha, 24 linhas.
- Caracteres 5 × 7.
- Vídeo normal, reverso e piscante.
- Controle pleno do curso.

Modo Gráfico
(baixa resolução)
- 40 h × 48 v ou 40 h × 40 v com 4 linhas de texto.
- 16 cores selecionáveis por Software.
- Comando específicos do Polysoft Basic para uso do Modo Gráfico: COLOR, PLOT, HLIN, VLIN, SCRN.

(alta resolução)
- 280 h × 192 v ou 280 h × 160 v com 4 linhas de texto.
- 6 cores selecionáveis por Software.
- Comandos específicos do Polysoft Basic para uso no modo gráfico: HCOLOR, HPLOT.
- Imagem do vídeo residente em 8 kbytes.

Memória
A memória dinâmica RAM é organizada em 3 incrementos de 16 kbytes cada um, num total de 48 kbytes totalmente disponíveis para o usuário. Esta memória pode ser aumentada conforme a necessidade do usuário adicionando placas de expansão. Possui também 10 kbytes de ROM para armazenamento do Polysoft Basic e 2 kbytes de ROM para o sistema monitor. Sistema de "Refresh" automático, completamente transparente. Memória rápida — tempo de acesso de 350 ns.

Entrada e Saída
O **Maxxi** inclui um teclado com padrão ASC II; interface para gravador cassete, vídeo e tv colorida sistema PAL-M; um conjunto de 8 conectores para a ligação de cartões controladores de periféricos e expansões; 3 entradas de 1 bit, 4 entradas analógicas para conexão de "joystick" e 4 saídas digitais de 1 bit.

Polysoft Basic
Possui características básicas do padrão Basic com técnicas de forma a propiciar máximo rendimento dos recursos de Software do produto.

Monitor
Gravado em ROM com 2 kbytes.

Periféricos e acessórios disponíveis
O **Maxxi** de concepção modular, cresce de acordo com os periféricos a ele incorporados, dentre os quais destacamos: TV comum, a cores ou preto & branco — Monitor profissional Polymax de 12'' com fosfatização verde — Unidade de drives de disquetes de 5¼'', organizado com 25 trilhas, 16 setores de capacidade de 256 bytes cada um — Unidade de gravador cassete — Interface serial para impressora — Impressora Polyprint 90 CPS de 80/132 colunas — Controladores de jogos — Interface serial para comunicação de dados — Modem — Placa de expansão de memória RAM para 64 kb — Placa de CPU Z-80 (sistema operacional CP/M) — Placa Videx expansora de vídeo de 40 para 80 colunas — modulador de R.F.

- Outros produtos da **Polymax:**
- **POLY 301 WP (Polyscriba)**
- **POLY 201 DP • POLY 105 DP**
- **POLY 201 DE • POLYNET**

MAXI

Polymax SISTEMAS E PERIFÉRICOS S.A.

Filiada à ABICOMP

MATRIZ: PORTO ALEGRE (RS) - Fone: 42-7833.
FILIAIS: ABC (SP) - Fone: 454-4922 - **BELO HORIZONTE** (MG) - **BRASÍLIA** (DF) - Fone: 225-1456 - **CURITIBA** (PR) - Fone: 233-6632 **PORTO ALEGRE** (RS) - Fone: 42-3311 - **RIO DE JANEIRO** (RJ) - Fone: 252-8274 - **SÃO PAULO** (SP) - Fone: 283-3722.

REDE DE REVENDEDORES:
BAURU (SP): Soma - 24-2558 - **BELÉM (PA):** Pam - 222-9772 - **BELO HORIZONTE (MG):** Compucity - 226-6336 - Computronics - 225-3305 - Julio Lobos - 225-6519 - **BRASÍLIA (DF):** Compushow - 273-2128 - GB - 242-6344 - MB - 226-5914 - Vídeo Service 248-6321 - **CAMPO GRANDE (MS):** DRL - 382-6487 - New Line - 624-5349 - **CAMPINAS (SP):** Microsystems - 51-3542 - Microtok - 32-3910 - **CAXIAS DO SUL (RS):** Digipampa - 221-4559 - **CURITIBA (PR):** Comicro - 224-5616 - Compustore - 232-1750 - Compusystem - 243-1731 - **DUQUE DE CAXIAS (RJ):** CPM - 771-0312 - **FORTALEZA (CE):** General Data - 226-2610 - **GOIÂNIA (GO):** Gendados - 224-5487 - **ITAJAÍ (SC):** Entec - 44-0244 - **JOINVILLE (SC):** Comicro - 22-5858 - **JUIZ DE FORA (MG):** Vermac - 212-3809 **LONDRINA (PR):** Comicro - 23-0065 - **MANAUS (AM):** CPD - 237-1793 - **MOGI DAS CRUZES (SP):** Runners House - 468-3779 - **NOVO HAMBURGO (RS):** Micromega - 93-4721 - **PASSO FUNDO (RS):** Digipampa - 312-3169 - **PELOTAS (RS):** CCS - 25-4139 **PIRACICABA (SP):** Sogemec - 34-2100 - **PORTO ALEGRE (RS):** Advancing - 26-8246 - DB - 22-5136 - Digital - 24-1411 - Informatic - 21-4189 - Microsis - 22-9782 - Processa - 26-0936 - **RIBEIRÃO PRETO (SP):** Compusys - 635-1195 - Khedi - 634-1715 - Memocards 636-0586 - **RECIFE (PE):** General Data - 222-0357 - Intertécnica - 341-2467 - Mekros - 224-3216 - **RIO DE JANEIRO (RJ):** Clap - 234-0214 - Computique - 267-1093 - Eldorado - 227-0791 - Garson - 252-2050 - Kristian - 252-9057 - Micromaq - 222-6088 - Microshow 264-5797 - SMC - 239-1345 - **SALVADOR (BA):** Sismac - 243-0991 - **SANTA MARIA (RS):** Digipampa - 221-6952 - **SANTA ROSA (RS):** Agnoleto - 512-1399 - **SANTOS (SP):** Kauffmann - 34-1476 - Sisper - 37-4705 - **SÃO JOSÉ DOS CAMPOS (SP):** Log - 22-7311 Siscom - 23-3752 - **SÃO LUIZ (MA):** MPA - 221-1755 - **SÃO PAULO (SP):** Audio - 220-2322 - Compushop - 212-9004 - Compute - 852-8290 - Computerland - 258-3954 - Data Magna - 272-2432 - Eletrogige - 881-6224 - Fotóptica - 852-2172 - Imarés - 61-4049 881-0222 - Infordata - 853-5740 - Mappin - 258-7311 - Microshop - 282-2105 - Prokura - 32-9776 - PSI - 531-9902 - Reflex - 883-2440 - Tiger - 212-9522 - **SOROCABA (SP):** Datamed - 33-3155 - **UBERLÂNDIA (MG):** Silógica - 234-4191 - **VOLTA REDONDA (RJ):** Bevorelli - 42-2175 - **VITÓRIA (ES):** Taborda - 222-5395.

* Apple é marca registrada de Apple Computer Inc.

cartas

Goiânia-GO, 19 de Setembro de 1983

Estimada Naila,

Com todo o respeito e admiração por essa equipe de "INTERFACE", desejo expressar minha felicidade em tentar pertencer a tão magnífica família. Para isso, semana passada, mandei meu pedido de assinatura juntamente com o vale postal de Cr$ 7.500,00.

Informo a nobre irmã que não entendo nada de eletricidade e muito menos de Informática. Ocorre que há um mês, mais ou menos, veio de Brasília, morar com minha família, um afilhado de 19 anos transferido para o CPD do Bradesco de Goiânia. Não me lembro que em algum momento eu e minha família houvéssemos tocado no assunto "Computador". Pois bem, em menos de um mês a linguagem do meu lar transformou-se totalmente. O rapaz trouxe um CP 200, e tudo mudou. É no café, no almoço, na janta, o tema é um só – INFORMÁTICA.

Consegui alguns números atrasados da revista INTERFACE e na nº 1 gostei dessa frase do Editor ". . . especial cuidado será dado aos novos adeptos da informática, e que sejam bem-vindos . . ." Obrigado.

Adorei o editorial da nº 2 e a homenagem singela a Francisco Oliveira, do qual nunca ouvi falar por ser eu um neófito de vossa família. Mas, minha sensibilidade sentiu o espírito da coisa, e orei.

Naila, meus parabéns pela Nota do Editor *da INTERFACE nº 5 e a felicidade nessa colocação: ". . . a posição de INTERFACE nesse problema é a nível de estímulo da criatividade nacional . . .". Sabemos que a criatividade brasileira é inigualável no mundo; o que nos falta é justamente estímulos.*

No editorial da nº 8, quando é abordado "os interesses escusos de tecnocratas que acusam a tecnologia nacional de obsoleta . . ." – Olha, Naila, o sangue me sobe à cabeça. Porque tenho um vasto conhecimento de como o poder econômico alienígena (via CIA) compra a peso de muito dólar esses "patriotas" – que aliás o inferno está cheio deles. E, para aqueles que se julgam insubstituíveis, uma frase do ex-Presidente Castelo Branco: "O cemitério está cheio de insubstituíveis".

Para quem quiser fazer uma pesquisa mais profunda sobre o assunto dos inimigos do Brasil, aliás, tem muitos que nasceram aqui, leiam as obras do professor universitário, jornalista e pesquisador de ciências políticas e sociais – Moniz Bandeira.

Prezada Naila, curto demais o seu Editorial e, quero parabenizá-la mais uma vez, em nome de toda a minha família (daqui), pelas suas colocações sóbrias e sensatas. Não tenho mais palavras para enaltecer essa equipe briosa da INTERFACE. Não nego também que tenha deixado de dar uma olhadinha nas outras revistas do ramo. Sempre que posso dou uma desfolhada nas co-irmãs de INTERFACE, apesar dos olhares pouco amistosos dos donos das bancas. E foi nessa pesquisa "lambe-lambe", nas bancas, que realmente descobri juntamente com os outros hobbistas que INTERFACE é sem sombra de dúvidas a melhor revista da atualidade em termos de Informática. O curso BASIC que vocês começaram na nº 10 é fenomenal, fora os demais cursos que também o são.

Amigo Wiles, tivemos um grande prazer e porque não dizer, emoção *com o recebimento de suas cartas. Eu e toda a minha equipe.*

A informática como ciência tem imenso prazer em acolhê-lo como um novo adepto e INTERFACE um especial prazer em tê-lo junto a sua família. Achamos, realmente, que seu esforço e de sua esposa certamente será recompensado e vocês alcançarão os melhores êxitos.

Em nosso primeiro aniversário, este foi um grande presente.

Abraços, Naila.

COM OS NOSSOS AGRADECIMENTOS

Parabéns pela excelente revista INTERFACE, tanto na qualidade de impressão quanto no conteúdo técnico.

Garet J. Cammer
Gerente de Produtos Eletrônicos
Indústrias Filizola S/A
São Paulo – SP

Sou leitor assíduo desta ótima revista e venho parabenizar-lhes pelo alto nível dos artigos publicados tanto em software quanto em hardware. A cada nova publicação, INTERFACE vem tornando-se elemento indispensável em termos de informação para a classe profissional na área de Processamento de Dados.

José Eduardo Ribeiro da Costa
São Paulo – SP

Gostaria de cumprimentar-lhes pela ótima revista que é INTERFACE. Das outras revistas sobre microcomputadores, hardware e software acho que INTERFACE é a melhor pois é mais profunda, sem que a literatura fique exclusivamente técnica.

Roberto Pinto Nazario
Campinas – SP

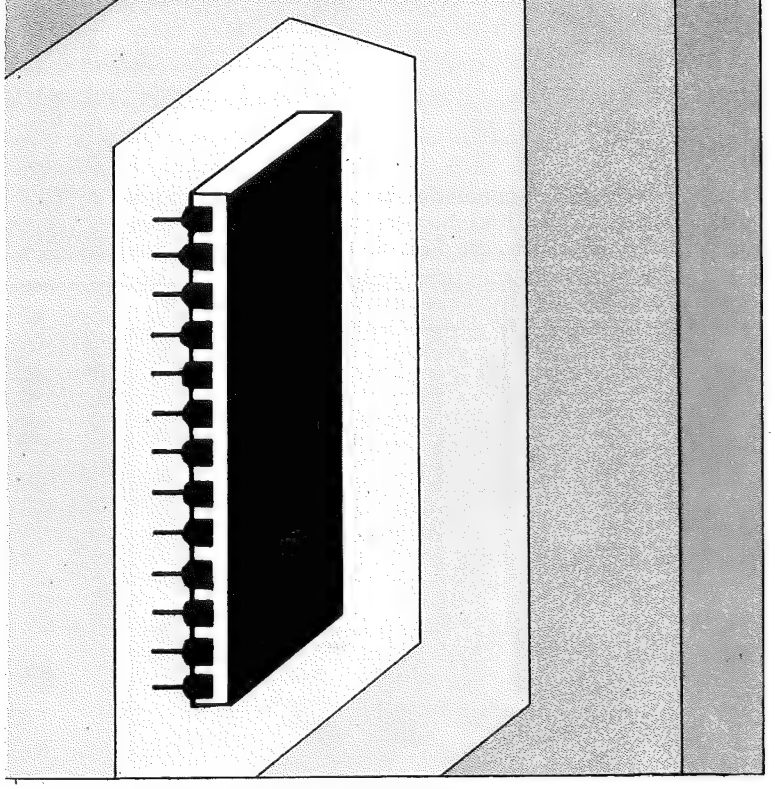
Aproveito a oportunidade para externar meus cumprimentos pelo excelente nível de qualidade que está sendo mantido na revista e faço votos que assim continue, pois, acho o Brasil muito deficiente neste tipo de publicação.

Mauro Lucio Carrusca
Engenheiro de Hardware
Quartzil Informática S.A.
Belo Horizonte – MG

Mais uma vez parabenizo-os pelo excelente trabalho mostrado na revista mensal INTERFACE. Tenho apreciado muito os exemplares e sentido uma notável preocupação de V. Sas no que se refere ao hardware.

No que se refere à parte de software, observei a meta na qual os senhores se propõem a publicar programas de grande utilidade.

Paulo Cesar Bonetti
Juiz de Fora – MG



CURSO Z-80

Por André Gil Rubens e Ney Acyr de Oliveira

12ª Lição

Nesta lição apresentaremos as instruções lógicas e aritméticas de 8 bits.

GRUPO DE INSTRUÇÕES LÓGICAS E ARITMÉTICAS DE 8 BITS

O grupo de *instruções lógicas e aritméticas de 8 bits* realiza operações de adição, subtração, "e" lógico, "ou" lógico, "ou" exclusivo, comparação e incrementação e decrementação de uma unidade.

As instruções deste grupo, por sua vez, classificam-se em dois subgrupos. O *primeiro subgrupo* envolve operações com dois operandos. Neste caso, um operando é sempre o acumulador e o outro pode ser especificado por *endereçamento imediato, registrador indireto,* através do par HL, *indexado* ou *registrador.*

O *segundo subgrupo* envolve operações com um operando, que pode ser especificado pelas mesmas técnicas de endereçamento do subgrupo 1, *exceto o imediato.*

A tabela 1 apresenta as *instruções lógicas e aritméticas de 8 bits,* com um resumo sucinto das suas principais características.

SUBGRUPO 1 — OPERAÇÕES COM DOIS OPERANDOS

As funções lógicas e aritméticas processadas pelo subgrupo 1 operam o acumulador e o outro operando especificado, e o resultado é sempre depositado no acumulador. Os bits de condição são atualizados em função do resultado da operação realizada.

Os mnemônicos das instruções do subgrupo 1 são compostos sempre do código da operação (ADD, ADC, SUB, SBC, AND, OR, XOR e CP) e do segundo operando, que varia em função da técnica de endereçamento usada; r, n, (HL), (IX + d) e (IY + d).

OPERAÇÕES ARITMÉTICAS

As operações aritméticas com operandos de 8 bits, no Z-80, são as de adição e as de subtração.

OPERAÇÕES DE ADIÇÃO

Nas operações de adição o registrador especificado é adicionado ao acumulador através da aritmética complemento a dois. Existem dois tipos de operações de adição: *sem o carry* e *com o carry.*

No primeiro caso, o conteúdo do acumulador é adicionado ao segundo operando e o resultado é depositado no acumulador.

No segundo caso, o conteúdo do acumulador é adicionado ao segundo operando e ao bit carry. E, este resultado é depositado no acumulador.

As operações de *adição com o carry* permitem a realização de operações de *adição de multiprecisão,* que estudaremos posteriormente.

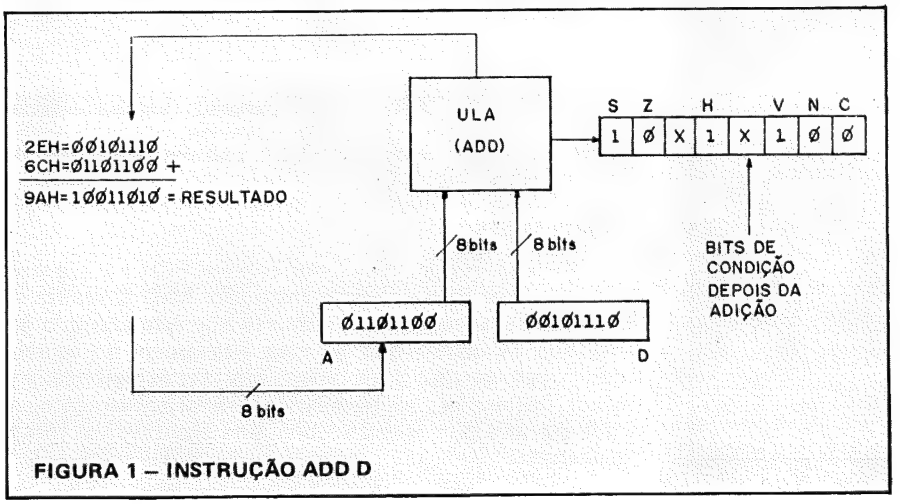


FIGURA 1 — INSTRUÇÃO ADD D

EXEMPLO # 1

Assuma que o registrador D contém 2EH e o acumulador contém 6CH, então, a instrução ADD D realiza a operação de adição, conforme mostra a figura 1.

EXEMPLO # 2

Assuma que o registrador C contém 3DH, o acumulador 42H e o carry-bit = 1, então, a instrução ADC C realiza a operação de adição conforme mostra

a figura 2.

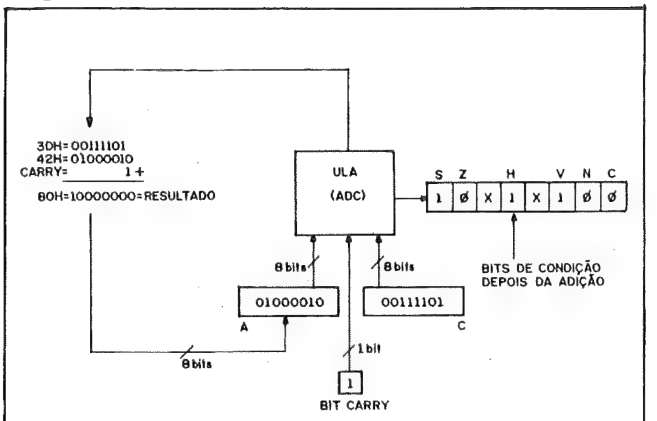


FIGURA 2 – INSTRUÇÃO ADC C

OPERAÇÕES DE SUBTRAÇÃO

Nas operações de subtração o conteúdo do registrador especificado é subtraído do conteúdo do acumulador na aritmética complemento a dois.

A ação do bit carry nas operações de subtração difere da operação de adição. Na subtração, se não for gerado "vai um" do bit mais significativo então o carry-bit é setado, indicando a ocorrência do "um emprestado" (borrow). Caso contrário, o carry-bit é resetado.

Da mesma forma que nas operações de adição, existem dois tipos de operação de subtração: *sem o borrow* e *com o borrow.*

Na subtração sem o borrow o segundo operando é subtraído do conteúdo do acumulador. Em contrapartida, na subtração com o borrow, do acumulador é subtraído o resultado da soma do segundo operando ao estado atual do bit carry.

As instruções de subtração com borrow são muito úteis na implementação de operações de *subtração de multiprecisão.*

EXEMPLO # 3

Assuma que o acumulador contém 3EH. Então, a instrução SUB A subtrai o acumulador dele mesmo, produzindo um resultado nulo. A figura 3 ilustra esta operação. Observe que como o "vai um" do bit de mais alta ordem é um, tratando-se de uma operação de subtração o bit carry fica setado.

Note que a instrução SUB A pode ser usada para resetar o carry-bit e zerar o acumulador.

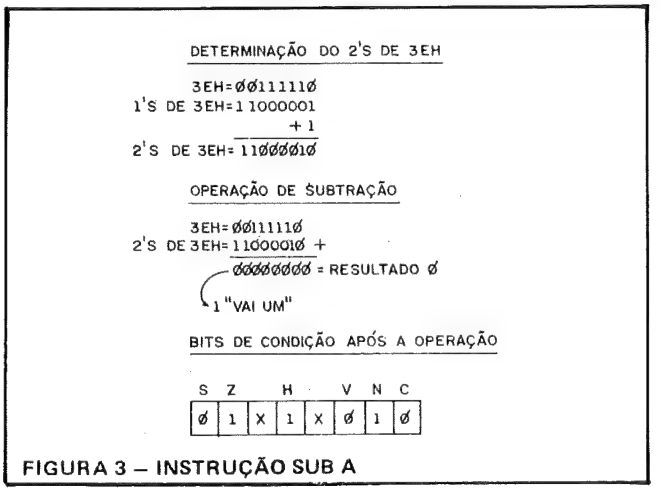


FIGURA 3 – INSTRUÇÃO SUB A

EXEMPLO # 4

Assuma que o conteúdo da posição de memória 1020 H contém 02H. O acumulador contém 04H e o carry bit = 1. Então, a instrução SBC A, (IX + 20H) realiza a operação de subtração conforme mostra a figura 4, sabendo-se, ainda, que no momento de sua execução IX = 1000H.

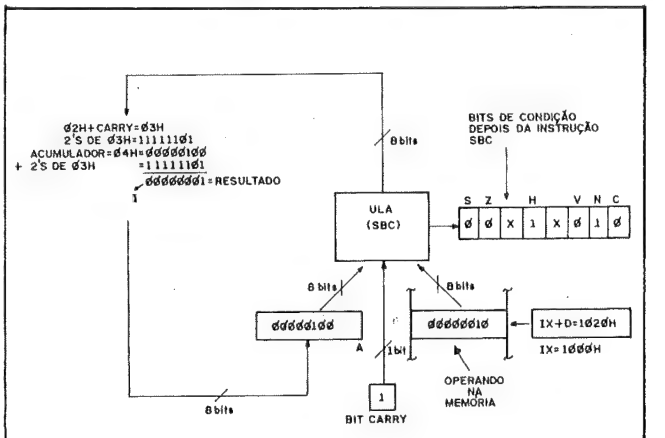


FIGURA 4 – INSTRUÇÃO SBC (IX + 20H)

OPERAÇÕES LÓGICAS

As três operações lógicas possíveis são o "e", "ou" e "ou exclusivo" lógicos, e a comparação. A tabela 2 mostra a simbologia destas operações, exceto a comparação, e a tabela 3 mostra as suas tabelas-verdades. A instrução de comparação será estudada separadamente.

TABELA 2
SIMBOLOGIA DAS OPERAÇÕES LÓGICAS

FUNÇÃO LÓGICA	MNEMÔNICO DA INSTRUÇÃO	SÍMBOLO DA OPERAÇÃO
"e"	AND	$\wedge$
"ou"	OR	$\vee$
"ou exclusivo"	XOR	$\oplus$

TABELA 3
TABELAS-VERDADES

OPERANDOS		RESULTADOS		
OPERANDO 1	OPERANDO 2	$\wedge$	$\vee$	$\oplus$
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

As funções lógicas no Z-80 operam os bits *de mesma ordem* em cada byte, realizando a operação bit a bit. Como em bit não afeta o seu adjacente, em conseqüência, não há "vai um" (carry-bit = 0).

EXEMPLO # 5

Assuma que o acumulador contém FCH e o registrador C contém 0FH, então a instrução AND C realiza as ações mostradas na figura 5. Observe que este exemplo garante que os quatro bits de mais alta ordem do acumulador fiquem zerados e, os quatro bits menos significativos fiquem inalterados; na terminologia de processamento de dados dizemos que os quatro bits mais significativos foram "mascarados".

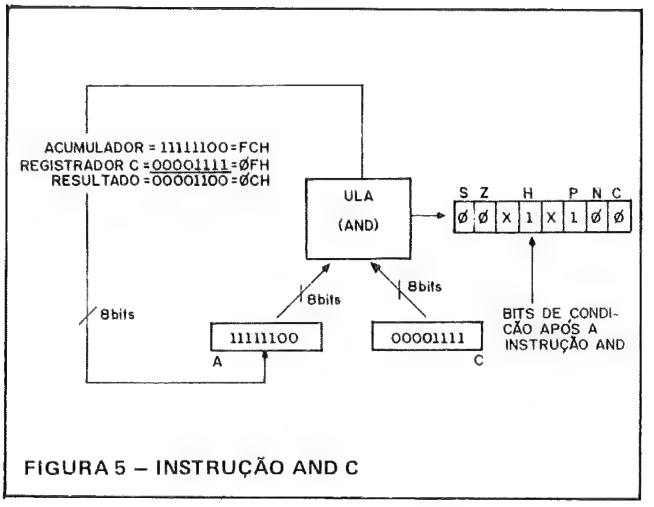


FIGURA 5 – INSTRUÇÃO AND C

De um modo geral, como o "e" lógico de um bit com "zero" dá "zero" o "e" lógico de "um" com qualquer bit mantém este bit inalterado. O *"e" lógico é usado para gerar grupos de bits.*

A operação de resetar ou setar bits mantendo os demais inalterados é denominada de "mascarar" (mask).

EXEMPLO # 6

Assuma que o registro D contém B5H, então a instrução OR A,0FH procede conforme mostra a figura 6. Observe neste exemplo que os quatro bits de mais baixa ordem do acumulador ficam setados e os quatro bits mais significativos ficam inalterados. Podemos dizer que os quatro bits mais significativos foram "mascarados".

De um modo geral, o "ou" lógico de um bit com "um" dá "um" e o "ou" lógico de "zero" com qualquer bit mantém este bit inalterado. O *"ou" lógico é usado para setar grupos de bits.*

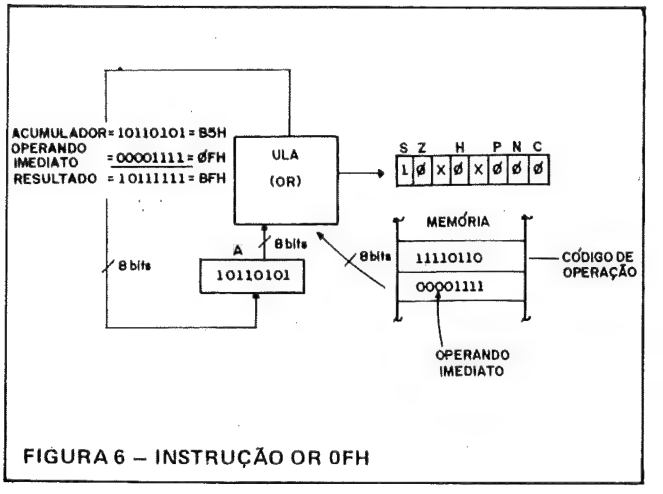


FIGURA 6 – INSTRUÇÃO OR 0FH

EXEMPLO # 7

Como o "ou" exclusivo de qualquer bit com ele mesmo produz um zero, então a instrução XOR A pode ser usada para gerar o conteúdo do acumulador.

EXEMPLO # 8

Como o "ou" exclusivo de "um" lógico com qualquer bit é o seu complemento (0 XOR 1 = 1 e 1 XOR 1 = 0), então, se o conteúdo do acumulador é FFH a instrução XOR B produz o complemento a um do registrador B no acumulador.

OPERAÇÃO DE COMPARAÇÃO

A instrução *CP* compara o segundo operando com o acumulador. Esta comparação se faz subtraindo internamente o operando do acumulador, deixando ambos inalterados, e atualizando os bits de condição em função do resultado.

Se Z = 1 significa que as quantidades são iguais, Z = 0 significa que as quantidades são diferentes.

Como se trata de uma operação de subtração, o bit carry fica setado se não houver "vai um" do bit mais significativo do resultado. Se os operandos forem números positivos, este resultado (C = 1) significa que o segundo operando é maior do que o acumulador. A figura 7 resume a análise dos bits Z e C, assumindo a comparação de números positivos.

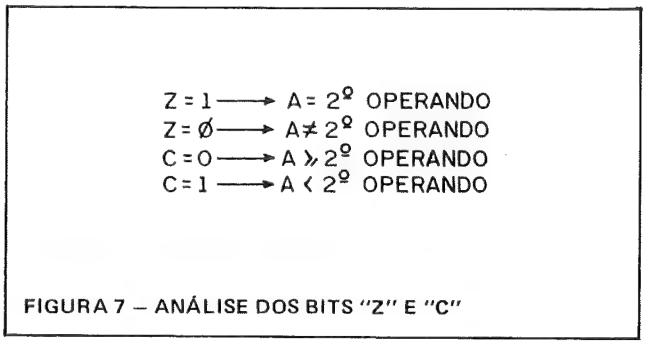


FIGURA 7 – ANÁLISE DOS BITS "Z" E "C"

EXEMPLO # 9

Assuma que o conteúdo do acumulador é 0BH e o registrador D contém 04H. Então, a instrução CP D realiza a operação de subtração interna mostrada na figura 8.

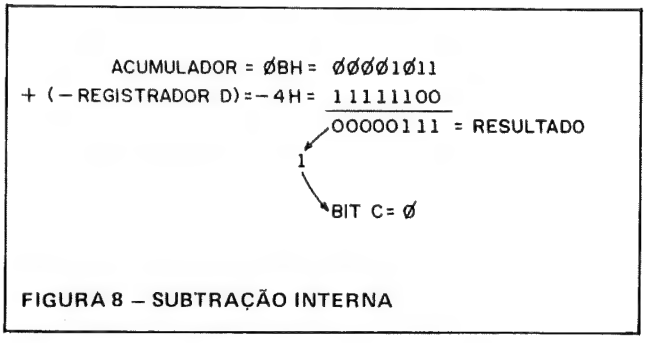


FIGURA 8 – SUBTRAÇÃO INTERNA

Após a subtração temos A = 0BH, D = 04H e o bit C = 0, indicando que o conteúdo de D é menor ou igual ao conteúdo do acumulador.

SUBGRUPO 2 – OPERAÇÕES COM UM OPERANDO

O *subgrupo 2* envolve as operações de incrementar e decrementar de uma unidade o conteúdo de um registrador interno do Z-80 ou de uma posição de memória. O resultado destas operações é depositado no próprio operando. Os bits de condição S, Z, H e V são atualizados em função do resultado. No entanto, *nas operações do subgrupo 2 o carry bit fica inalterado.*

Os mnemônicos das instruções do subgrupo 2 são compostos sempre do código da operação (INC e DEC) e do operando (r, (HL), (IX + d) e (IY + d)).

EXEMPLO # 10

Se o registrador H contém 79H, então, após a execução da instrução INC H passa a ter 7AH. A figura 9 mostra o procedimento da instrução INCH para este exemplo.

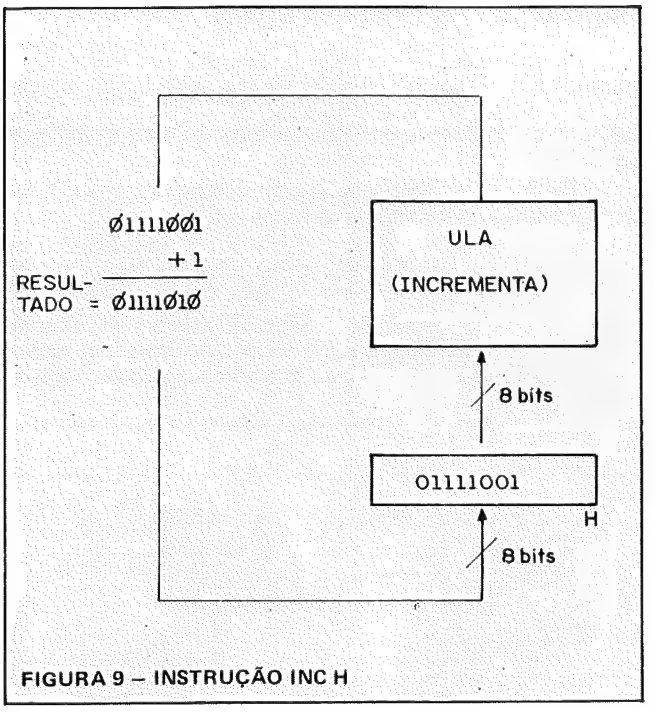


FIGURA 9 – INSTRUÇÃO INC H

EXEMPLO # 11

Assuma que o registrador IY contém 5000H e que a posição de memória 5010H contém 04H. Então, após a execução da instrução DEC (IY + 10H) a posição de memória 5010H passa a ter 03H.

SOMA DE MULTIPRECISÃO

A instrução *ADC* pode ser usada para somar números que precisem de mais de um byte para sua representação. Considere a soma de dois números hexadecimais sem sinal da figura 10. Esta soma pode ser feita no Z-80, somando-se primeiramente os dois bytes menos significativos e em seguida o "vai um" resultante à parcela seguinte da operação.

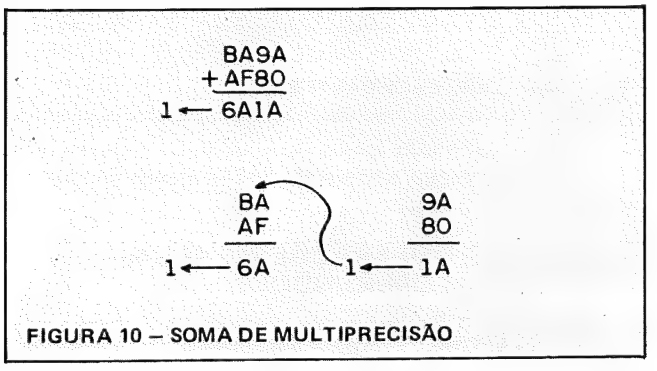


FIGURA 10 – SOMA DE MULTIPRECISÃO

Observe que desta forma podemos adicionar parcelas com qualquer número de dígitos. A isto denominamos de *adição de multiprecisão.*

EXEMPLO # 12

O programa apresentado neste exemplo soma dois números de 16 bits armazenados em bytes consecutivos de memória. A primeira parcela tem o byte menos significativo no endereço PARC1 e a segunda parcela em PARC2. O resultado da operação fica armazenado em PARC1. A lógica do programa está mostrada na figura 11, e o programa na linguagem Assembly do Z-80 na figura 12.

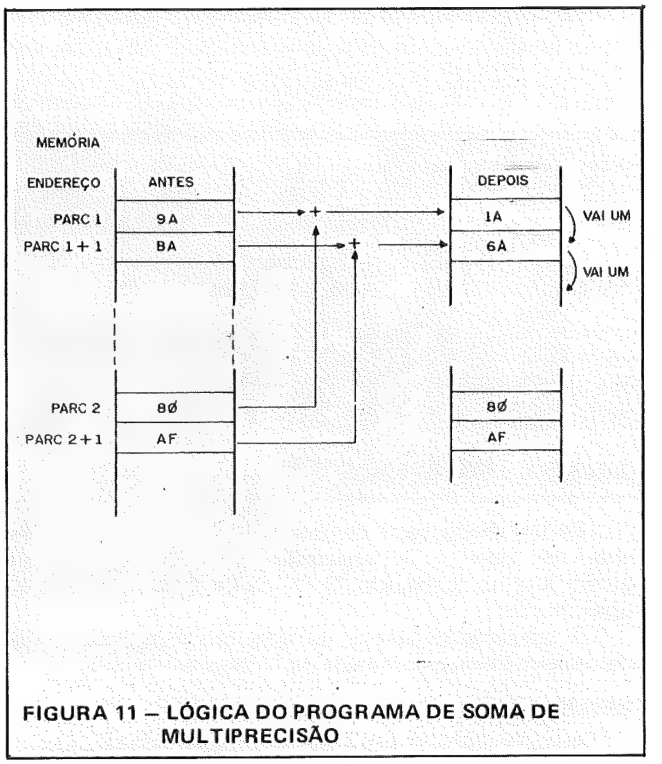


FIGURA 11 – LÓGICA DO PROGRAMA DE SOMA DE MULTIPRECISÃO

```
ORG     1ØØH

LD      BC, PARC1       ; BC PONTEIRO PARC1
LD      HL, PARC2       ; HL PONTEIRO PARC2
LD      A,(BC)          ; CARREGA PARC1
ADD     (HL)            ; SOMA MENOS SIGNIFICATIVA
LD      (BC),A          ; GUARDA RESULTADO
INC     L               ; ATUALIZA PONTEIROS
INC     C
LD      A,(BC)          ; CARREGA PARC1+1
ADC     (HL)            ; SOMA MAIS SIGNIFICATIVA
LD      (BC),A          ; GUARDA RESULTADO
HALT
FIM
```

FIGURA 12 – PROGRAMA DE SOMA DE MULTIPRECISÃO

Observe que ao atualizarmos os ponteiros com as instruções INC L e INC C, não há passagem de "vai um" da parte menos significativa para a mais significativa de cada ponteiro.

SUBTRAÇÃO DE MULTIPRECISÃO

A partir do exemplo # 11, para se implementar um programa de subtração de multiprecisão basta substituir a instrução ADD por SUB e a instrução ADC por SBC.

COMPARAÇÃO COM O 8080

O microprocessador 8080 tem as mesmas funções aritméticas e lógicas de 8 bits que o Z-80.

No entanto, o Z-80 tem as vantagens de poder designar operações pelo endereçamento indexado e de gerar os bits de condição V e N.

TABELA 1 – INSTRUÇÕES LÓGICAS E ARITMÉTICAS DE 8 BITS

MNEMÔNICOS 8080/85	MNEMÔNICOS Z-80	OPERAÇÃO SIMBÓLICA	FLAGS C	Z	P/V	S	N	H	Código de Máquina Binário 76	543	210	Nº de Bytes	Nº de M-Ciclos	Nº de T-Ciclos	SUB-GRUPO
OPERAÇÕES ARITMÉTICAS DE 8 BITS COM DOIS OPERANDOS															
ADDr	ADD r	A ← A + r	↕	↕	V	↕	0	↕	10	000	r	1	1	4	
ADI n	ADD n	A ← A + n	↕	↕	V	↕	0	↕	11	000	110	2	2	7	
									←	n	→				
ADD M	ADD (HL)	A ← A + (HL)	↕	↕	V	↕	0	↕	10	000	110	1	2	7	
–	ADD (IX + d)	A ← A + (IX + d)	↕	↕	V	↕	0	↕	11	011	101	3	5	19	
									10	000	110				
									←	d	→				
–	ADD (IY + d)	A ← A + (IY + d)	↕	↕	V	↕	0	↕	11	111	101	3	5	19	
									10	000	110				
									←	d	→				1
ADC, ADI	ADC s	A ← A + s + CY	↕	↕	V	↕	0	↕		001					
SUB, SUI	SUB s	A ← A – s	↕	↕	V	↕	1	↕		010					
SBB, SBI	SBC s	A ← A – s – CY	↕	↕	V	↕	1	↕		011					
OPERAÇÕES LÓGICAS DE 8 BITS COM DOIS OPERANDOS															
ANA, ANI	AND s	A ← A ∧ s	0	↕	P	1	0	1		100					
ORA, ORI	OR s	A ← A ∨ s	0	↕	P	1	0	1		110					
XRA, XRI	XOR s	A ← A ⊕ s	0	↕	P	1	0	1		101					
CMP, CPI	CP s	A – s	↕	↕	V	↕	1	↕		111					
OPERAÇÕES ARITMÉTICAS COM UM OPERANDO															
INR r	INC r	r ← r + 1	●	↕	V	↕	0	↕	00	r	100	1	1	4	
INR M	INC (HL)	(HL) ← (HL) + 1	●	↕	V	↕	0	↕	00	110	100	1	3	11	
–	INC (IX + d)	(IX + d) ← (IX + d) + 1	●	↕	V	↕	0	↕	11	011	101	3	6	23	
									00	110	100				
									←	d	→				2
–	INC (IY + d)	(IY + d) ← (IY + d) + 1	●	↕	V	↕	0	↕	11	111	101	3	6	23	
									00	110	100				
									←	d	→				
DCR	DEC d	d ← d – 1	●	↕	V	↕	1	↕			101				

NOTAS:

1 – "r" simboliza qualquer um dos registradores de 8 bits: B (000), C (001), D (010), E (011), H (100), L (101), A (111).

2 – "s" simboliza r, n, (HL), (IX + d) e (IY + d), conforme mostrado nas instruções ADD.

3 – Os bits no interior dos retângulos das instruções do subgrupo 1 substituem o $\phi\phi\phi$ nas instruções ADD.

4 – O "d" na instrução DEC simboliza r, (HL), (IX + d) e (IY + d), conforme mostrado nas instruções INC.

5 – Para se obter os códigos de máquina binários das instruções DEC, basta substituir os bits $1\phi\phi$ no interior dos retângulos nas instruções INC por $1\phi1$.

6 – A letra V indica que o flag P/V fornece o overflow do resultado, e a letra P a paridade.

7 – V = 1 indica a ocorrência de overflow
P = 1 indica que a paridade do resultado é par

8 – Notação dos flags: ● = não afetado, 0 = resetado, 1 = setado.
X = indefinido.
↕ = o flag é afetado em função do resultado da operação.

A&N

Consultoria em hardware e software de microcomputadores

Projetos de Interfaces
Controle de processos
Teleprocessamento

DESENVOLVIMENTO DE SOFTWARE
Aplicações Comerciais
Aplicações em tempo real

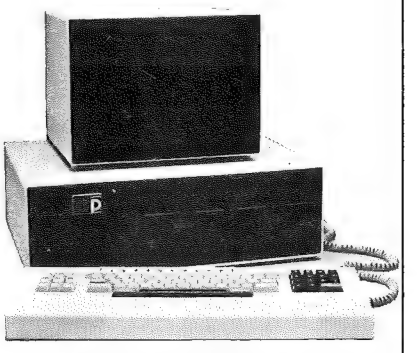
André Gil Rubens – CREA 30911 – D
Ney Acyr R. Oliveira – CREA 38324 – D Tel.: (021) 248-8105

novos produtos

ED-P O MICROCOMPUTADOR PESSOAL DA EDISA

Com 2 disquetes de 5" 1/4 e monitor de 24 linhas por 80 colunas, o microcomputador pessoal ED-P tem também teclado separado, ligado por cabo múltiplo flexível espiralado, contendo caracteres alfanuméricos, funções especiais e teclado numérico.

O *ED-P* utiliza um microprocessador Z-80A de 4MHz e pode ser expandido desde sua configuração básica de 64 Kbytes RAM até 1024 Kbytes, e ainda em 1984 poderá receber também um microprocessador de 16 bits, aumentando assim consideravelmente sua velocidade de operação.



COENCISA TAMBÉM NA III FEIRA DE INFORMÁTICA

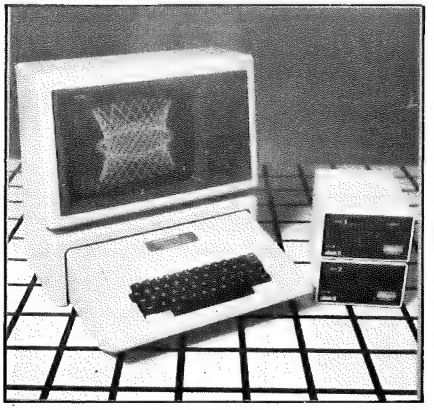
A linha TVC de terminais de vídeo e a linha TCD de terminais portáteis de coleta de dados foram apresentadas na III Feira de Informática.

A linha de terminais de vídeo oferece os modelos TVC 1000 e TVC 1300. Na concepção desta linha foi atribuída ênfase ao *design* do produto, considerando-se os aspectos estéticos e ergonômicos. Os produtos estarão disponíveis para entrega a partir do segundo semestre de 1984. A estimativa de preço de comercialização para o usuário final é de 330 ORTN's para o TVC 1000 e 380 ORTN's para o TVC 1300. Será dada garantia de um ano contra problemas de fabricação.

ELPPA II PLUS – O NOVO MEMBRO DA FAMÍLIA APPLE

A empresa Victor do Brasil Eletrônica Ltda lançou o microcomputador *ELPPA II PLUS,* compatível em hardware e software com o microcomputador APPLE II.

O *ELPPA II PLUS* utiliza o microprocessador 6502, capacidade de memória RAM de 48 Kbytes expandível, capacidade de memória EPROM de 12 Kbytes, software residente em EPROM (intérprete e compilador para Basic-ELPPASOFT), capacidade de texto (960 caracteres), capacidade gráfica, capacidade de cores, periféricos (disco flexível de 5 1/4", impressora paralela, CP/M e monitor) e capacidade de expansão.



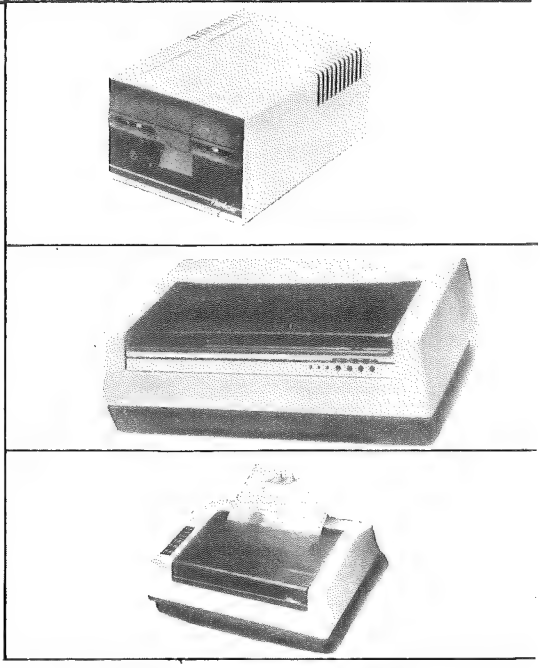
ELEBRA E O LANÇAMENTO DE NOVOS PRODUTOS NA III FEIRA DE INFORMÁTICA

A Elebra Informática S.A apresentou ao público uma série de produtos novos, que ampliam e complementam sua linha, para melhor atender às necessidades do mercado. Entre eles citamos: a *impressora serial Mônica,* a *impressora serial Alice* e as *unidades de disco flexível.*

A *impressora serial Mônica (EI6010)* tem baixo custo e grande durabilidade, com cabeça de impressão de 300 milhões de caracteres, destinada principalmente ao mercado de micros pessoais. Pode ser alimentada com formulários contínuos ou papel-carta. Na configuração básica, a *Mônica* é uma impressora de 100 cps, com 80 posições de impressão de 10 cpp ou 132, a 16,7 cpp. Pode, porém, ser configurada para diferentes aplicações, através da fácil adição de módulos.

A *impressora serial Alice (EI9050)* tem velocidade de 200 cps, interface serial RS232-C ou paralela Centronics/ Dataproducts, 136 colunas, caracteres normais e expandidos e controle eletrônico por microprocessador. Utiliza formulários com até 16,75 polegadas de largura e 7 vias.

Na área de discos foram apresentadas as unidades de *disco flexível 9408, de 5 1/4",* com cabeça dupla e capacidade de 0,5 Mbyte, e o *Horácio,* unidade de disco flexível acondicionada em caixa metálica, com placa de interface, cabos e manuais de instalação, já prontas para serem usadas com micros pessoais tipo Apple e tipo TRS-80, de diversos fabricantes brasileiros.



A linha de terminais portáteis visa a coleta de dados de campo com armazenamento local, para posterior transmissão a um computador central via aparelho telefônico comum.

Os exemplares de linha serão acondicionados em valise de mão, especialmente projetada. São alimentados por baterias recarregáveis, com carregador embutido.

SPECTRUM/ MICROENGENHO 2

Pouco mais de dezoito meses após o lançamento do seu primeiro microcomputador pessoal, o MicroENGENHO, a Spectrum participou da III Feira de Informática lançando o MICROENGENHO 2, um equipamento que cria uma nova filosofia no sistema de compatibilidades.

O MICROENGENHO 2 é o "profissional para uso pessoal" que reúne num só equipamento as características mais positivas dos melhores microcomputadores do mundo (Apple e IBM PC).

TCOLOR E SYSDATA III

A Sysdata Eletrônica lançou o *TCOLOR*, microcomputador estritamente pessoal para competir com o TK83 e 85, destinado a uma faixa de mercado de menor poder aquisitivo (configuração mínima Cr$ 280.000,00 e configuração máxima Cr$ 380.000,00).

Foi lançado também o SYSDATA III, microcomputador profissional com versões de 48 Kbytes e 64 Kbytes. O teclado é destacável da CPU que, por sua vez, apresenta incorporado ao seu gabinete, embutidos, os drives de disco.

O sistema operacional é o CP/M 2.2, LDOS ou TRS/DOS, podendo acionar até quatro drives de disco de 5 1/4" de dupla densidade ou até três drives de 8" de dupla face e dupla densidade. Possui interface opcional tipo winchester, interface RS 232, saída para impressora paralela ou serial, vídeo de 25 X 80 colunas, teclado profissional e teclado numérico reduzido. O preço de lançamento na configuração de 48 Kbytes e dois drives de 5 1/4" é a partir de Cr$ 2.300.000,00.

RINGO – O NOVO MEMBRO DA LÓGICA SINCLAIR

A Ritas do Brasil lançou o microcomputador *R-470*, "RINGO", destinado a uma faixa de usuários que vai desde os iniciantes até os proprietários de pequenas empresas e profissionais liberais.

O "RINGO" utiliza o microprocessador Z80A, capacidade de memória RAM de 16 Kbytes expandível, capacidade de memória ROM expandível de 8 Kbytes, interfaces para gravador cassete (2400 bps), Joystick, modem (1200 bauds), vídeo (canal 3), teclado tipo QWERTY com 49 teclas, teclas de edição com repetição automática, periféricos (interface para máquina de escrever elétrica como impressora, discador de telefone, gravador de memória EPROM, banco de memória com retenção de dados à bateria, sintetizador de sons) e linguagens BASIC e códigos de máquina Z-80.

Traduções técnicas Inglês/Português Português/Inglês exclusivas para área de informática

- Traduções exclusivas para a comunidade de informática
- Pioneiros no setor
- Manuais e publicações diversas
- Gráficos
- Supervisão técnica de profissionais

atuantes, analistas e consultores
- "Linguagens de alto nível" aliadas à qualidade/ velocidade de entrega
- Informações pelos tels. (021) 264-6392/ 264-7391/228-2798

• • •

DISCOVER
TRADUÇÕES
End.: Rua Professor Gabizo, 225
Tijuca Rio de Janeiro - RJ Cep. 20271

CURSO BASIC I

- Em 8 aulas práticas
- 4.000 a aula com apostila e professor
- Um microcomputador por aluno
- Horário de acordo com o seu tempo disponível
- Início imediato

Barry FEIRA PERMANENTE DE MICROCOMPUTADORES LTDA.

MICROCOMPUTADORES
- Em 3 pagamentos sem acréscimo
- O melhor preço à vista ou até 24 meses
- Os lançamentos da III Feira Internacional de Informática
- Novidades em jogos
- Fitas aplicativas para empresas
- CLUBE DO MICRO - Se você não tem Microcomputador poderá usar os nossos Suprimentos, revistas e livros
- Software por encomenda

Tel.: (011) 570-1555 - SP
Rua Domingos de Moraes, 407 - Próximo Metrô Ana Rosa

calendário

COMPARAÇÃO E SELEÇÃO DE MICROCOMPUTADORES

A Compucenter realizará, em São Paulo, de 5 a 7 de dezembro um curso destinado a profissionais envolvidos em avaliação e seleção de equipamentos, bem como, a todas as pessoas interessadas em conhecer mais os microcomputadores. Este curso apresentará a metodologia para avaliar e selecionar a melhor alternativa de equipamento. Todas as características dos micros brasileiros, incluindo preços, serão debatidas.

Informações pelos telefones: (011) 255-9662 ou 255-5988.

ADP – DIVISÃO DE ENSINO EM COMPUTAÇÃO

A ADP Systems, Divisão de Ensino, está oferecendo os seguintes cursos: *Programação de Sistemas* (180 horas), *Análise de Sistemas* (288 horas), *Operação de Sistemas* (168 horas), *Assembler* (45 horas) e *Basic* (45 horas). Em alguns dos cursos são oferecidos estágios práticos. Maiores informações na Rua Santa Isabel, nº 305 – Tel.: (011) 223-7511 – SP.

CURSOS/SCI

A SCI promove em dezembro os seguintes cursos: *Análise Estruturada para Sistemas com Banco de Dados* (6 a 9/12 – São Paulo), *Bancos de Dados Relacionais: Fundamentos, Impacto e Uso* (7 a 9/12 – RJ), *Protocolo X-25* (7 a 9/12 – RJ), *Como Determinar Prazos Certos em Desenvolvimento de Sistemas* (7 a 9/12 – São Paulo), *Informática para Usuários* (8 a 9/12 – RJ).

Informações à Rua Jardim Botânico, 635 – 8º andar, Tel.: (021) 294-7438 – Rio de Janeiro. E, em São Paulo, à Av. Paulista, 2001 – Grupo 1020, Tel.: (011) 289-0099.

FLASH COURSES

A Eldorado Computadores e Sistemas Ltda estará iniciando uma série de "flash courses" em *Basic I, Basic Avançado* e *Utilização de Aplicativos,* em janeiro de 84. As crianças terão turmas especiais com dois alunos por micro.

Inscrições e informações à Rua Visconde de Pirajá, 351 – Loja 213/214, Ipanema – Rio.

CÂNDIDO MENDES/INFORMÁTICA

O Centro Cultural Cândido Mendes está promovendo cursos na área de Informática. A era do microcomputador de uso pessoal chegou para ficar. Sua utilização começa a fazer parte de nossa vida diária, a exemplo do que ocorre em outros países. A mão-de-obra necessária deve ser formada a curto prazo, e em qualidade compatível com o potencial de um microcomputador. Atendendo a essa demanda ora iniciada, o Centro Cultural Cândido Mendes e a Datamicro Informática Ltda oferecem os seguintes cursos: *Introdução aos Microcomputadores e Programação Basic.*

Informações e inscrições à Rua Visconde de Pirajá, nº 547, Loja 211, Tel.: (021) 274-1042 – RJ.

ERRATA DO SIMULADOR DE VÔO (2ª PARTE)

Em virtude de alguns leitores não terem conseguido identificar os códigos errados na tabela 1, publicado na errata da revista número 11, página 39, publicamos a seguir os códigos corretos com uma nova forma de localização.

Lembramos aos leitores a necessidade de substituir a segunda linha do programa, que cria uma área para a rotina em assembler, por *"2REM"*, a fim de que o programa não entre em loop.

POSIÇÃO A ALTERAR NA TABELA 1

Endereço	*CONTEÚDO Publicado*	*Correto*	*Localização na Tabela 1*
17980	37	36	Linha 17964 – Coluna 17
18270	08	09	Linha 18264 – Coluna 7
19321	CB	CD	Linha 18314 – Coluna 8
18548	0F	8F	Linha 18539 – Coluna 10
18612	C1	21	Linha 18589 – Coluna 24
18627	19	16	Linha 18614 – Coluna 14
18771	5A	4A	Linha 18764 – Coluna 8

DESCRIÇÃO DE LINHAS INCOMPLETAS NA LISTAGEM

5641 IF C(19) + X (BN) > 7.5 AND C (3)
< 1500 THEN GOTO 3020
5040 PRINT "VOCÊ BATEU CONTRA O " ; X$

OBS.: Caso tenha dúvida consulte a revista nº 5, página 36.

Linhas com descrição de símbolos para alterar

7090 alterar . . . (25) SPACE para (26) SPACE

8020: na 2ª linha alterar . . . (7) SPACE para (8) SPACE
na 15ª linha alterar:
. . . SPACE + SHIFT J + (6) SPACE + SHIFT R + (2) . . .
para
(2) SPACE + SHIFT R + SPACE + SHIFT
J + (6) SPACE . . .
na 23ª linha alterar:
. . . + SPACE + O (letra) + . . . para
+ *(2)* SPACE + O (letra) + . . .
na 13ª linha alterar:
último número entre parênteses é (6)

8106 na 2ª linha alterar:
1/2 por C

8120 na 7ª/8ª linhas
retirar . . . SHIFT E + SHIFT 8 + (30) SPACE

8130 na 6ª linha
retirar (2) SPACE

Interface

NOVEMBRO/83

SUPLEMENTO DO PEQUENO USUÁRIO DE MICROCOMPUTADOR

- Curso de Programação em Linguagem Basic
- Dicas de Software
- Glossário de Termos Técnicos
- Programas Aplicativos
- Dicas de Hardware
- I Comunidade de Projetistas de Hardware e Software – COPHAS

CURSO DE PROGRAMAÇÃO EM LINGUAGEM BASIC

4ª Lição

Por José Arthur da Rocha

Na lição anterior mostramos como implementar o algoritmo de definição de um problema na forma de um programa na linguagem BASIC. Os exercícios propostos na lição três estão resolvidos a seguir.

No primeiro exercício utilizamos os algoritmos propostos na primeira lição e solucionados na segunda, usando a seqüência de Fibonacci. Para o fluxograma desenvolvido na figura 1 da segunda lição, onde era pedido o primeiro número maior que 1000 e a sua posição na seqüência de Fibonacci, foi implementado o programa seguinte:

```
 10  P = 0
 20  U = 1
 30  CONT = 2
 40  S = P + U
 50  CONT = CONT + 1
 60  IF S> 1000 THEN GOTO 100
 70  P = U
 80  U = S
 90  GOTO 40
100  PRINT S
120  PRINT CONT
130  STOP
```

Observe o leitor que, conforme o comentado ao final da terceira lição, suprimimos o comando LET nas atribuições. Voltamos a lembrar que se o seu microcomputador não aceita tal simplificação o comando LET deverá ser mantido em todas as atribuições feitas dentro do programa.

Comentando ainda sobre as versões mais recentes do BASIC, elas permitem que dois ou mais comandos sejam colocados numa mesma linha, desde que separados por um símbolo gráfico, normalmente o dois pontos (:). Assim, eles continuam a ser executados seqüencialmente na ordem em que estiverem na linha que, por sua vez, será executada na ordem seqüencial das linhas dentro do programa. Isto traz uma grande vantagem no BASIC, principalmente pela redução dos números de linha utilizados no programa.

ALGORITMO PARA TRANSFORMAR POLEGADAS EM CENTÍMETROS

Sabendo que 1 polegada = 2,54 centímetros, só nos falta o valor a ser transformado. E talvez até queiramos transformar várias medidas em polegadas para centímetros aproveitando o mesmo algoritmo. Este, portanto, deverá aceitar o dado em polegadas, fornecer o seu equivalente em centímetros e estar pronto a receber outros dados ou terminar. Tal algoritmo é representado pelo fluxograma da figura 1.

Observe que o fluxograma da figura 1 tem duas entradas. A primeira delas está bem visível, caracterizada pela caixa de entrada de dados, quando é pedido o valor em polegadas. A segunda está camuflada pelo teste realizado no final do fluxograma, quando se pergunta se ainda existe outro dado a ser convertido. A resposta ao teste deve vir de fora, isto é, deve ser dado pela pessoa que está executando o algoritmo e, portanto, também é um dado de entrada.

Para a comunicação do operador com o algoritmo desenvolvido no computador, o BASIC utiliza a instrução INPUT (variável), onde a variável pode ser numérica ou alfanumérica (string). O programa abaixo pode ser utilizado na implementação do fluxograma da figura 1.

```
10  INPUT VP
20  VC = VP * 2.54
30  PRINT VC
40  PRINT "EXISTE OUTRO VALOR?"
50  INPUT T$
60  IF T$ = "SIM" THEN GOTO 10
70  STOP
```

Quando o computador executa a linha 10 do programa, ele pára e espera que o operador dê a entrada de dados que está sendo pedida. A máquina indica que está esperando esta entrada colocando um ponto de interrogação (?) na tela do vídeo. O programa continuará a ser executado depois que o operador entrar com o dado pelo teclado do terminal e apertar o RETURN (< RET >). Este dado deve ser compatível com a variável que está sendo pedida (no exemplo ela deve ser numérica), sob pena do computador acusar erro.

A execução continuará escrevendo na tela o valor VC da conversão e enviando a mensagem da linha 40. Na linha 50 ela dará uma nova parada ao encontrar o INPUT T$. Agora a variável de entrada é uma string que, em resposta a pergunta, será SIM ou NÃO. Um destes valores ou strings será atribuído a T$ via teclado pelo operador. Entrando com um deles e pressionando RETURN, o computador testa T$ na linha 60 e decide se deverá pedir novo dado (GO-TO 10) em caso de resposta afirmativa ou acabar o programa (70), no caso contrário.

A instrução INPUT permite que uma mensagem seja colocada antes da variável cujo valor está sendo requisitado, indicando para o operador que o algoritmo está esperando que ele dê entrada de um valor e a que variável ele será atribuído. A linha 10 do programa poderia ser então:

```
10 INPUT "ENTRAR VALOR EM POLEGADAS"; VP
```

Observe que a mensagem deve aparecer primeiro que a variável, que é separada desta por um ponto e vírgula (;), e que deve estar entre aspas (" ") pois é uma string. Esta mensagem aparecerá no vídeo e logo após ela o ? indicativo da parada à espera da entrada de dados.

Poderíamos também substituir as linhas 40 e 50 por:

```
50 INPUT "EXISTE OUTRO VALOR" ; T$
```

A instrução INPUT permite ainda que se dê entrada de um número maior de valores na mesma instrução. No exemplo da primeira lição, lembrado anteriormente, o algoritmo pedia a entrada de quatro notas, que poderia ser dado por:

```
10 INPUT "ENTRE COM 4 NOTAS"; N1, N2, N3, N4
```

Observe que as variáveis são separadas por vírgulas. Portanto, devem ser dados no exemplo quatro valores separados por vírgulas.

SAÍDA DE DADOS

Em todos os exemplos colocados até agora necessitamos dar uma saída de dados, que é a resposta da execução do algoritmo. Esta saída era normalmente obtida na linguagem BASIC através da instrução PRINT.

Veremos agora mais algumas características dessa instrução.

Pegando ainda como exemplo o fluxograma da figura 1, vemos que a linha 30 do programa BASIC que o implementa não coloca exatamente o que está indicado na caixa de saída de dados. Nesta, é dada uma mensagem que acompanha o valor VC. Tal mensagem também pode ser colocada na instrução PRINT. Assim, a linha 30 do programa poderia ser trocada para:

```
    30 PRINT "VALOR EM CENTÍMETROS = " ; VC
ou  30 PRINT "VALOR = " ; VC ; "CENTÍMETROS"
```

Observe que no primeiro caso é impressa a string exatamente como ela aparece entre aspas. O ponto e vírgula (;) após a string permite que a impressão continue na mesma linha e justaposta ao último caractere impresso. Portanto, o valor de VC será impresso logo após o espaço em branco deixado depois do sinal de igualdade (=) da string. No segundo caso ocorre algo semelhante e após VC ainda será colocada a palavra CENTÍMETROS, mostrando assim que valores numéricos e/ou strings podem ser impressos em qualquer ordem dentro do comando PRINT e serão colocados lado a lado se estiverem separados por ponto e vírgula (;).

Anteriormente, quando um PRINT era executado, o próximo que ocorria escrevia seu valor na linha de baixo do vídeo. Por exemplo:

```
100  PRINT 1
110  PRINT 2
120  PRINT 3
```

Ao ser executado o programa da linha 100 aparecerão os números 1, 2 e 3, um em cada linha. Isto acontece porque após cada instrução não existe nenhum outro caractere. Se houver, por exemplo, após o caractere 2 da linha 110 um ponto e vírgula (;) e considerando a definição do ponto e vírgula (;) dentro do PRINT apresentada acima, serão impressos o caractere 1 numa linha e os caracteres 2 e 3 justapostos na linha de baixo.

Se dentro do comando PRINT as variáveis e/ou valores a serem impressos estiverem separados por vírgulas (,) será obedecida uma tabulação automática, isto é, os valores serão impressos em colunas do vídeo pré-estabelecidas. Verifique isto no seu microcomputador!

Com o que foi visto até agora sobre entrada e saída de dados podemos, a título de exemplo, implementar o programa BASIC para o algoritmo do "cálculo da aprovação de alunos através da média de 4 provas", desenvolvido na primeira lição, cujo fluxograma é reapresentado na figura 2, um pouco melhorado.

Como exemplo, implementaremos o fluxograma da figura 3 da segunda lição utilizando esta técnica. Naquele exercício era pedida a soma dos 30 primeiros termos da seqüência de Fibonacci. O programa-solução é o seguinte:

```
10  P = 0 : U = 1 : CONT = 2 : SOMA = 1
20  S = P+U : CONT = CONT + 1 : SOMA = SOMA + S
30  IF CONT = 30 THEN GOTO 50
40  P = U : U = S : GOTO 20
50  PRINT SOMA : STOP
```

Observe que se tivéssemos colocado um comando em cada linha utilizaríamos 11 linhas, ocupando um maior espaço na memória do computador. Fizemos o mesmo programa com apenas 5 linhas utilizando vários comandos na mesma linha.

Neste programa temos duas observações a fazer. A primeira é que separamos a atribuição "S = P + U" das quatro anteriores colocadas na linha 10. A razão é lógica, pois o algoritmo pede uma volta justamente àquele comando, conforme o indicado no fluxograma. Se tal não acontecesse, as três atribuições da linha 20 poderiam ser colocadas também na linha 10, separadas, evidentemente, pelo dois pontos (:). A quantidade de comandos possíveis numa linha só depende do número de caracteres permitidos na linha de cada máquina e, obviamente, se o microcomputador aceita esta técnica.

A segunda observação é que poderíamos não ter utilizado a linha 50, bastando para isso substituirmos a linha 30 por:

```
30  IF CONT = 30 THEN PRINT SOMA : STOP
```

Observe novamente a técnica dos comandos colocados na mesma linha. No entanto, aqui ela tem uma conotação ligeiramente diferente. Nesta linha, se a expressão for verdadeira, isto é, se CONT = 30, ambos os comandos colocados depois do THEN serão executados naquela ordem. E o programa pararia ali, pois o segundo comando é o STOP. Caso contrário, se CONT < 30, o programa desprezará o que está após o THEN e passará para a linha 40, pela própria definição do comando IF. Com isto "economizamos" mais uma linha.

Neste ponto voltamos a lembrar ao leitor que esta técnica não existe em algumas versões do BASIC e, se este for o caso de sua máquina, ela não poderá ser utilizada. O programa então deverá ser desenvolvido como o primeiro apresentado, isto é, com um comando por linha.

Continuando a resolução dos exercícios da lição 3, o segundo nos coloca 3 situações onde se pergunta para que valores das variáveis serão impressos os asteriscos (*). Vejamos cada uma delas.

a) IF X = 0 OR Y ⩽ 10 THEN PRINT "*"

O asterisco será impresso se a expressão composta for verdadeira e, pela definição do operador lógico OR, isto acontecerá se pelo menos uma delas o for. Assim, o asterisco será impresso se x = 0 ou se y ⩽ 10. Em outras palavras, ele não será impresso se x ≠ 0 e y > 10. Observem a relação entre os operadores lógicos OR (ou), AND (e) e NOT (negação):

X = 0 OR y ⩽ 10 corresponde a
NOT (NOT (x = 0) AND NOT (y ⩽ 10)) que corresponde a
NOT (x ≠ 0 AND y > 10)
Verifique!

b) IF A + B = 3 AND A = B OR A > B THEN PRINT "*"

Consideremos esta expressão composta por partes. Primeiramente temos uma operação AND entre as expressões simples "A + B = 3" e "A = B". Pela definição do operador AND, teremos:

A + B = 3 AND A = B ⇒ A = B = 1,5

Juntando agora esta expressão a outra expressão simples (A > B) através do operador OR, teremos que a expressão composta será verdadeira sempre que A for maior que B ou, se iguais, tem que valer 1.5, para que seja impresso o asterisco.

A hierarquia dos operadores lógicos como o de outras funções pode variar de uma versão para outra. No entanto, para a maior parte delas a operação AND tem maior hierarquia que a OR. A hierarquia do NOT é maior que os dois anteriores, embora tenha pouca aplicação no BASIC.

c) IF I ⩾ 10 AND I < 5 THEN PRINT "*"

A primeira expressão simples será verdadeira para valores de I ⩾ 10 e a segunda somente para I < 5 e conseqüentemente são exclusivas, isto é, nunca serão verdadeiras simultaneamente, o que torna a expressão composta sempre falsa e o asterisco jamais será impresso.

ENTRADA DE DADOS

Na maioria dos exemplos vistos até agora em nosso curso os valores atribuídos às variáveis ou dados eram gerados pelo próprio algoritmo. Isto era feito através de atribuições diretas do valor (por exemplo: U = 1) ou por expressões do tipo S = P + U, como no ciclo de geração de dados da seqüência de Fibonacci.

No entanto, muitas vezes desejamos realizar alguma "tarefa" que envolva dados aos quais possamos atribuir qualquer valor, a qualquer tempo. Na primeira lição vimos um algoritmo para o cálculo da média de 4 notas para aprovação de alunos, onde essas notas eram "entradas" para o fluxograma. Vejamos um outro exemplo:

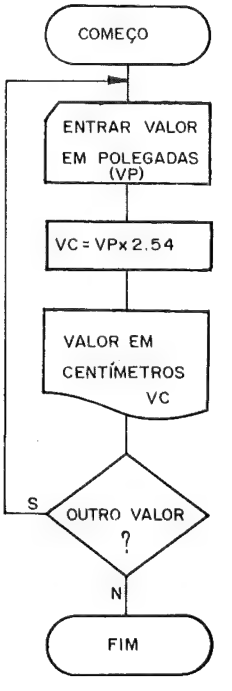


FIGURA 1 – FLUXOGRAMA QUE TRANSFORMA UMA MEDIDA EM POLEGADAS PARA CENTÍMETROS

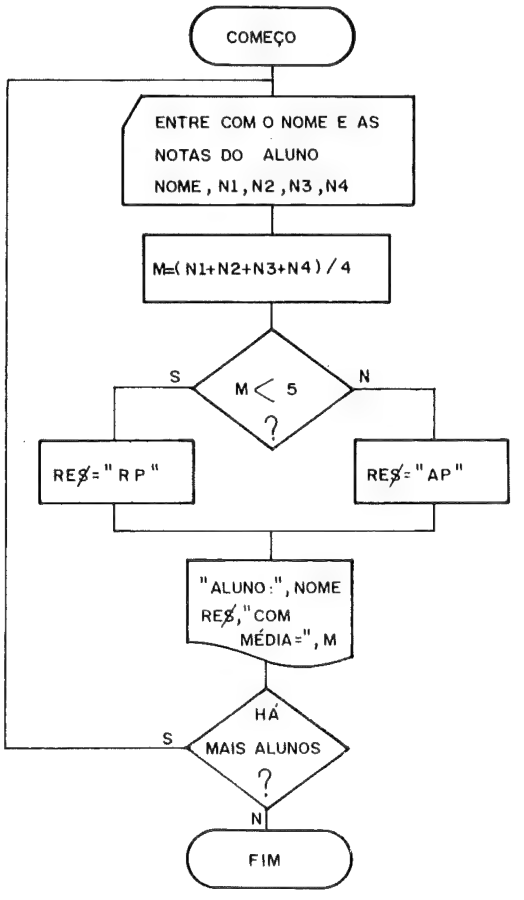


FIGURA 2 — FLUXOGRAMA PARA O CÁLCULO DA APROVAÇÃO DE ALUNOS ATRAVÉS DA MÉDIA DE 4 PROVAS

Observe que o primeiro teste realizado no fluxograma tem duas alternativas, como sempre, e depois de executados qualquer um dos caminhos, retorna-se ao mesmo ponto na caixa de saída de dados. Poderíamos traduzir em palavras aquele teste da seguinte forma:

SE M < 5 então RE$ = "RP" do contrário RE$ = "AP"

que, em inglês seria:
IF M < 5 THEN RE$ = "RP" ELSE RE$ = "AP"

que seria a instrução IF completa no BASIC

Ao ser executada é testado o valor da expressão M < 5. Se a verdadeira, é executado o que está depois do THEN (RE$ = "RP") e o programa continua na linha de baixo. Se ela é falsa não será executado o que está após o THEN e sim o que está após o ELSE (RE$ = "AP") e o programa continua na linha de baixo. Em resumo, neste tipo de instrução IF será executado o que está entre o THEN e o ELSE se a expressão for verdadeira; se falsa, é executado o que está após o ELSE. Após um dos dois casos será executada a linha de baixo, caso as instruções que estão após o THEN ou após o ELSE não façam um desvio incondicional (GO TO).

Assim, o fluxograma da figura 2 pode ser implementado com o programa abaixo:

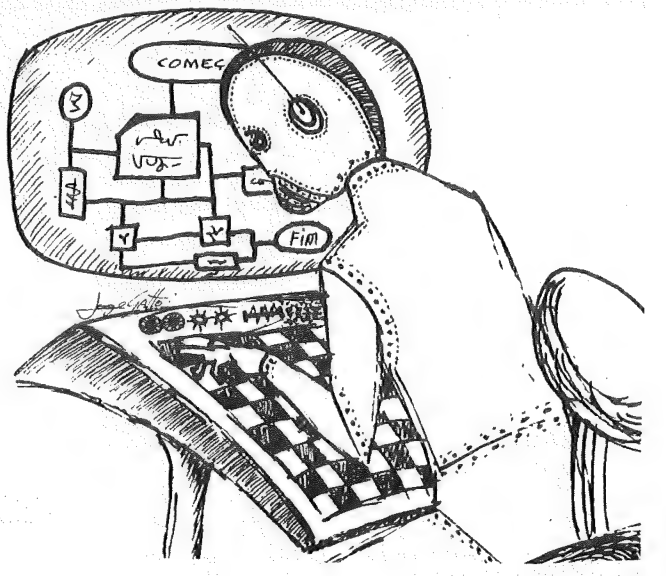
```
10  INPUT "ENTRE COM O NOME E AS NOTAS
    DO ALUNO" ; NOME$, N1, N2, N3, N4
20  M = (N1 + N2 + N3 + N4) /4
30  IF M < 5 THEN RE$ = "RP" ELSE RE$ = "AP"
40  PRINT "ALUNO:"; NOME$; "FOI" ; RE$ ;
    "COM MEDIA = " ; M
50  INPUT "HÁ MAIS ALUNOS" ; T$
60  IF T$ = "SIM" THEN GOTO 10
70  STOP
```

SUMÁRIO

Vimos nesta lição mais algumas diferenças que pode haver entre algumas versões do BASIC, apresentando algumas técnicas existentes nas mais recentes que facilitam a programação como, por exemplo, comandos colocados na mesma linha separados por dois pontos (:).

Além disso, apresentamos as principais instruções utilizadas no BASIC para entrada (INPUT) e para saída (PRINT) de dados, mostrando neste último algumas técnicas de formatação dos dados a serem impressos.

Aproveitamos aqui para lembrar ao leitor que verifique como são implementadas no seu microcomputador (caso o tenha) as técnicas aqui apresentadas.



Exercícios

• Faça o fluxograma e o programa BASIC que calcule as raízes da equação abaixo:

$$Ax^2 + Bx + C = 0$$

sabendo que os valores de A, B e C são entradas do algoritmo para que ele possa ser repetido várias vezes se o operador assim o desejar.

Atenção: dê as mensagens "RAÍZES IMAGINÁRIAS" caso isto aconteça e "DADOS REJEITADOS" caso A = 0 e a equação, portanto, não for do segundo grau.

Referências

• GOTTFRIED, B. S. — "Programming with BASIC", McGraw Hill, 2ª Edição, 1982.

• DGT-100 — Digibasic Manual Digitus.

CONTAS A PAGAR / RECEBER NO TK82-C, CP-200, TK85, SINCLAIR OU SIMILARES

Por Francisco José D. Figueiredo

COMENTÁRIOS:

- Digite o programa.
- Execute-o (RUN).
- O programa pedirá a data de hoje. Informe-a, começando pelo dia, o mês e finalmente o ano.

Obs.: A digitação do ano deverá ser padronizada; ou entra-se com o ano completo (ex. 1983) ou entra-se somente com as duas últimas posições (ex. 83). Após a escolha do processó, este não poderá ser mudado pois ocasionará erro no cálculo de dias.

```
 1 REM ***... (23) ... ***
 2 REM ****CONTAS A PAGAR ***
 3 REM ***... (23) ... ***
 4 REM AUTOR: FRANCISCO JOSÉ D. FIGUEIREDO
 5 REM ***... (23) ... ***
 6 DIM I (12)
 7 LET T$ = "0000310590901201511
   81212243273304334"
 8 LET Y = 0
 9 FOR X = 1 TO 36 STEP 3
10 LET Y = Y + 1
11 LET I (Y) = VAL T$ (X TO X + 2)
15 NEXT X
20 LET N = 0
30 DIM A$ (100, 30)
40 DIM A (100, 5)
55 GOSUB 60
57 GOTO 65
60 CLS
61 PRINT AT 2,4; " ***... (24) ... ***"
62 PRINT TAB 4; " **** CONTAS A PAGAR ****"
63 PRINT TAB 4; " ***... (24) ... ***"
64 RETURN
65 PRINT AT 7,5; "INFORME DATA DE HOJE"
66 GOSUB 1100
67 IF E THEN GOTO 65
68 LET D = M
69 GOSUB 1255
70 GOSUB 60
73 PRINT AT 11,20; "PROCURA = 0"
75 PRINT TAB 20; "LISTA = 1"
80 PRINT TAB 20; "30 DIAS = 2"
85 PRINT TAB 20; "60 DIAS = 3"
90 PRINT TAB 20; "APÓS 60 = 4"
93 PRINT TAB 20; "VENCIDO = 5"
95 PRINT TAB 20; "INCLUI = 6"
100 PRINT TAB 20; "ALTERA = 7"
105 PRINT TAB 20; "EXCLUI = 8"
110 PRINT TAB 20; "FIM = 9"
115 INPUT P
120 CLS
130 IF P = 0 THEN GOSUB 190
135 IF P = 1 THEN GOSUB 300
140 IF P = 2 THEN GOSUB 400
145 IF P = 3 THEN GOSUB 500
150 IF P = 4 THEN GOSUB 600
153 IF P = 5 THEN GOSUB 650
154 IF P = 6 THEN GOSUB 800
155 IF P = 7 THEN GOSUB 700
160 IF P = 8 THEN GOSUB 950
165 IF P = 9 THEN GOSUB 1000
```

```
170 GOTO 70
190 LET M$ = "NOME NÃO ENCONTRADO"
200 PRINT "DIGITE O NOME"
205 INPUT X$
210 LET K = LEN X$
213 LET E = 0
215 FOR Y = 1 TO N
218 IF A$ (Y, 1) = " *" AND P < > 6 THEN GOTO 225
220 IF X$ = A$ (Y, TO K) THEN GOTO 234
225 NEXT Y
227 IF P = 6 AND Y < 101 THEN RETURN
228 CLS
229 IF P < > 6 THEN PRINT , , X$
230 PRINT , , " ***"; M$; "***"
231 GOSUB 1255
232 LET E = 1
233 RETURN
234 IF P = 6 THEN RETURN
235 CLS
240 LET L = 0
242 LET T = A (Y, 2) – D
245 GOSUB 1600
252 GOSUB 1300
254 IF E THEN GOTO 225
255 RETURN
300 LET LI = – 1000
305 LET LS = 1000
315 LET S$ = "TOTAL GERAL"
320 GOSUB 1500
390 RETURN
400 LET LI = 0
405 LET LS = 30
410 LET S$ = "SUBTOTAL ATÉ 30 DIAS"
480 RETURN
500 LET LI = 31
505 LET LS = 60
510 LET S$ = "SUBTOTAL 30/60 DIAS"
515 GOSUB 1500
580 RETURN
600 LET LI = 61
605 LET LS = 1000
610 LET S$ = "SUBTOTAL > 60 DIAS"
615 GOSUB 1500
620 RETURN
650 LET LI = – 1000
655 LET LS = – 1
660 LET S$ = "SUBTOTAL VENCIDO"
665 GOSUB 1500
670 RETURN
700 LET M$ = "VERIFIQUE NOME CORRETO"
705 PRINT "QUAL A FICHA A ALTERAR"
```

• Em caso de digitação de uma data incorreta, o programa voltará ao terceiro desses itens.

• < NEW LINE > P/CONTINUAR significa aperte NEW LINE para continuar o programa, então, aperte a tecla mencionada.

• Será exibido o menu do programa; escolha a opção 9 (FIM). Esta opção é utilizada para gravar o programa e os dados em fita.

• "PREPARE O GRAVADOR" significa colocar o gravador no modo para gravação.

• Após a gravação do programa a execução voltará a pedir a data de hoje.

• Caso você não deseje prosseguir desligue o computador.

• Caso contrário, informe a data e escolha a opção do menu.

EXPLICAÇÃO DAS OPÇÕES DO MENU:

• PROCURA — Procura determinada conta mediante o nome.

• LISTA — Efetua listagem (em tela) de todas as contas arquivadas, emitindo o total no final.

• 30 DIAS — Efetua listagem de contas a vencer den-

```
710  GOSUB 200
715  IF E THEN RETURN
720  PRINT AT 8,0; "QUAL O ITEM A ALTERAR?"
722  PRINT ,,
     NOME = 1
     VENCIMENTO = 2
     VALOR = 3
     N. BANCO = 4
     JUROS DIA = 5"
724  INPUT F
726  IF F < > 1 THEN GOTO 736
728  PRINT , "NOVO NOME"
730  INPUT A$ (Y)
732  PRINT A$ (Y)
734  GOTO 775
736  IF F < > 2 THEN GOTO 746
738  PRINT "NOVO VENCIMENTO"
740  GOSUB 1100
742  IF E THEN RETURN
743  LET A (Y, 2) = M
744  LET A (Y, 1) = VAL T$
745  GOTO 775
746  IF F < > 3 THEN GOTO 756
748  PRINT , , "NOVO VALOR CR$ ";
750  INPUT A (Y, 3)
751  PRINT A (Y, 3)
754  GOTO 775
756  IF F < > 4 THEN GOTO 764
758  PRINT , , "NOVO N. BANCO";
760  INPUT A (Y, 4)
762  PRINT A (Y, 4)
763  GOTO 775
764  IF F < > 5 THEN GOTO 724
766  PRINT , , "NOVO JUROS";
768  INPUT A (Y, 5)
769  PRINT A (Y, 5)
775  PRINT , , " **** ALTERADO **** "
780  IF A (Y, 3) = 0 THEN LET A$ (Y) = " * "
790  IF A (Y, 3) = 0 THEN PRINT ,, "VALOR = 0
     TAMBÉM ** EXCLUÍDO **"
794  GOSUB 1255
795  RETURN
800  LET M$ = "ARQUIVO LOTADO"
810  LET X$ = " *."
820  GOSUB 210
825  IF E THEN RETURN
830  PRINT "NOME"
835  INPUT A$ (Y)
840  PRINT A$ (Y)
845  PRINT "DATA VENCIMENTO"
851  GOSUB 1100
```

```
854  IF E THEN GOTO 845
855  LET A (Y, 2) = M
857  LET A (Y, 1) = VAL T$
865  PRINT "VALOR CR$";
870  INPUT A (Y, 3)
872  PRINT A (Y, 3)
880  PRINT "N. BANCO";
885  INPUT A (Y, 4)
890  PRINT A (Y, 4)
892  PRINT "JUROS DIA (JRS/D CR$)";
894  INPUT A (Y, 5)
895  PRINT A (T, 5)
900  LET N = N + 1
905  PRINT , , " **** INCLUÍDO ****"
908  GOSUB 1255
910  RETURN
950  LET M$ = "VERIFIQUE NOME CORRETO"
952  IF N = 0 THEN RETURN
955  GOSUB 200
958  IF E THEN RETURN
960  LET A$ (Y) = " * "
965  PRINT AT 10,0; " ****EXCLUÍDO ****"
970  GOSUB 1255
990  RETURN
1000 PRINT AT 10,0; "PREPARE O GRAVADOR
     PARA GUARDAR O ARQUIVO"
1005 PRINT "QUANDO PRONTO APERTE
     NEW LINE"
1010 INPUT U$
1012 SAVE "PAGAR"
1015 PRINT , , , , " OK . ARQUIVO
     GRAVADO"
1020 GOSUB 1255
1030 GOTO 55
1100 PRINT " DIA";
1105 INPUT DX
1110 PRINT DX
1115 PRINT "MÊS";
1120 INPUT MX
1125 PRINT MX
1130 PRINT "
1135 INPUT AX
1140 PRINT AX
1152 LET E = 0
1154 IF DX < 1 OR DX > 31 OR MX < 1 OR
     MX > 12 THEN GOTO 1250
1156 IF DX > 30 AND MX < > 1 AND MX < >
     3 AND MX < > 5 AND MX < > 7 AND MX < > 8
     AND MX < > 10 AND MX < > 12 THEN GOTO 1250
1158 IF MX = 2 AND DX > 28 THEN GOTO 1230
1166 LET T$ = " "
```

tro de no máximo 30 dias, juntamente com o respectivo subtotal.

- 60 DIAS – Efetua listagem de contas a vencer entre 31 e 60, emitindo o respectivo subtotal.
- APÓS 60 – Idem o item acima, com relação às contas acima de 60 dias.
- VENCIDO – Idem o item acima, com relação às contas atrasadas.
- INCLUI – Efetua inclusões de contas.
- ALTERA – Efetua alterações nas contas.
- EXCLUI – Efetua exclusões de contas.
- FIM – Gravação do arquivo.

Todas as listagens contém nome, data de vencimento, quantidade de dias para a data de vencimento, valor, juros, total, banco e juros ao dia. Sendo que juros total e juros ao dia estão somente presentes em contas a pagar quando a referida conta estiver atrasada. O total há pouco referido é o valor dos juros diários multiplicado pela quantidade de dias em atraso, somado com o valor da conta.

Ao carregar o programa (LOAD " " ou LOAD "PAGAR" ou LOAD "RECEBER") ele iniciará automaticamente.

```
1167  IF DX 10 THEN LET T$ = T$ + "0"
1168  LET T$ = T$ + STR$ DX
1169  IF MX 10 THEN LET T$ = T$ + "0"
1170  LET T$ = T$ + STR$ MX
1172  IF AX 10 THEN LET T$ = T$ + "0"
1175  LET T$ = T$ + STR$ AX
1180  LET M = I (MX) + DX
1185  IF MX > 2 AND AX/4 = INT (AX/4)
      THEN LET M = M + 1
1190  LET M = M + INT (365.25 (AX-1)) + 1
1220  RETURN
1230  IF (AX/4 = INT (AX/4)) AND
      DX < 30 THEN GOTO 1166
1252  PRINT AT 20,0; "DATA INVÁLIDA"
1253  LET E = 1
1255  PRINT AT 21,0; "NEW LINE P/CONTINUAR"
1257  LET L = 0
1260  INPUT U$
1265  CLS
1270  RETURN
1300  PRINT AT 21,0; "E " " ESTE ? (S/N)"
1305  INPUT Z$
1310  IF Z$ <> "N" AND Z$ <> "S" THEN
      GOTO 1305
1315  PRINT AT 21,0;"
1320  IF Z$ = "S" THEN LET E = 0
1325  IF Z$ = "N" THEN LET E = 1
1330  RETURN
1500  IF N = 0 THEN RETURN
1505  LET S = 0
1510  LET L = 0
1515  FOR Y = 1 TO N
1520  IF A$ (Y, 1) = " * " THEN GOTO 1545
1525  LET T = A (Y, 2) – D
1530  IF T < LI OR T > LS THEN GOTO 1545
1533  LET J = 0
1535  GOSUB 1600
1540  LET S = S + A (Y, 3) + J
1545  NEXT Y
1550  GOSUB 1255
1553  CLS
1565  PRINT AT 10,0; S$; AT 11,0; "CR$"; S
1570  GOSUB 1255
1575  RETURN
1600  LET L = L + 5
1605  IF T < 0 THEN LET L = L + 2
1610  IF L > 19 THEN GOSUB 1255
1615  PRINT
1620  PRINT A$ (Y)
1625  LET T$ = STR$ (A (Y, 1)
1630  LET C = 1
1635  IF LEN T$ < 6 THEN LET C = 0
```

```
1640  PRINT "VENC.: "; T$ (1 TO C + 1)
      ;"/"; T$ (C + 2 TO C + 3); "/"; T$ (C + 4 TO C + 5);
      TAB 16; "("; ABS T; "D)"; TAB 24;
1645  IF T = 0 THEN PRINT "HOJE"
1650  IT > = 0 THEN GOTO 1680
1655  PRINT "VENCIDO"
1660  LET J = A (Y, 5) * ABS T
1680  IF T > 0 THEN PRINT
1684  PRINT "VALOR CR$"; A (Y, 3)
1685  IF T > = 0 THEN GOTO 1710
1690  PRINT "JUROS CR$"; J
1705  PRINT "TOTAL CR$"; A (Y, 3) + J
1720  PRINT "BANCO"; A (Y, 4); TAB 12;
      "JRS/D CR$"; A (Y, 5)
1725  RETURN
```

Observação: Nas linhas 1, 3 e 5 o número entre parênteses indica o número de vezes que o caractere (*) deverá ser editado.

Relação de linhas para alterar ou excluir, transformando o programa de contas a pagar em contas a receber.

Linha 2 alterar para:

```
2     REM ***CONTAS A RECEBER **
```

Linha 40 alterar para:

```
40    DIM A (100, 4)
```

Linha 62 alterar para:

```
62    PRINT TAB 4; " *** CONTAS A RECEBER ***"
```

Linha 722 alterar para:

```
722   PRINT , ,
      NOME = 1
      VENCIMENTO = 2
      VALOR = 3
      N. BANCO = 4"
```

Linha 756 alterar para:

```
756   IF F <> 4 THEN GOTO 724
```

Eliminar as seguintes linhas:

763, 764, 766, 768, 769, 892, 894, 895.

Linha 1012 alterar para:

```
1012  SAVE "RECEBER"
```

IMPLEMENTAÇÃO DA FUNÇÃO SLOW NO MICROCOMPUTADOR NEZ-8000

Um dos maiores problemas enfrentados pelo possuidor do NEZ-8000 é ocasionado quando digita-se qualquer tecla e a televisão perde o sincronismo horizontal. Este fato ocorre pois a geração dos sinais de vídeo e sincronismo, a decodificação do teclado e o processamento são realizados pelo microprocessador Z-80. Devido a este fato, quando o microprocessador está executando uma das funções citadas, as outras ficam sem atendimento. Por isso, ao ser digitada uma tecla, o microprocessador vai atendê-la, esquecendo-se naturalmente do sincronismo do vídeo (INTERFACE número 1, página 6).

DESCRIÇÃO DO CIRCUITO

O objetivo deste circuito, figura 1, é sincronizar o vídeo enquanto a CPU atende a digitação de uma tecla. Para tanto, a chave do circuito é o transistor T2, que tem sua base ligada à saída $\overline{HALT}$ (pino 18) do Z-80, seu emissor à entrada $\overline{NM1}$ (pino 17) e o coletor à entrada $\overline{WAIT}$ (pino 24). Recordemos (vide curso do Z-80, INTERFACE número 4) que a entrada $\overline{WAIT}$ tem o objetivo de sincronizar memórias ou dispositivos de E/S, lentos, com o Z-80, e que enquanto esta entrada for mantida em nível "0", o Z-80 fica "parado" aguardando que o dispositivo de E/S, lento, responda a sua solicitação de leitura ou escrita. A entrada $\overline{NM1}$ é uma entrada de interrupção ativada pela transição negativa, ou seja, transição do nível "1" para o nível "0". Esta linha é sempre reconhecida ao final da instrução em execução, forçando o Z-80 a iniciar sua execução pelo endereço 0066H. O conteúdo do contador de programa é automaticamente colocado em uma pilha externa a fim de que o usuário possa retornar ao programa interrompido. Porém, repetidos ciclos de $\overline{WAIT}$ podem evitar o término da instrução em execução. A saída $\overline{HALT}$ é uma saída que é colocada no nível "0", após a execução da instrução HALT. A instrução HALT coloca o Z-80 no estado de "espera", que basicamente é utilizada no final de um programa, após todas as instruções terem sido executadas, ou quando é necessário permanecer com o Z-80 "parado", aguardando um pedido de interrupção. Assim sendo, a saída $\overline{HALT}$, quando não está ativada, chaveia a base do transistor T2, permitindo que o sinal da saída do CI 25, pino 11, esteja presente nas entradas $\overline{NM1}$ e $\overline{WAIT}$ do Z-80, sincronizando o vídeo enquanto o Z-80 atende a uma solicitação de leitura.

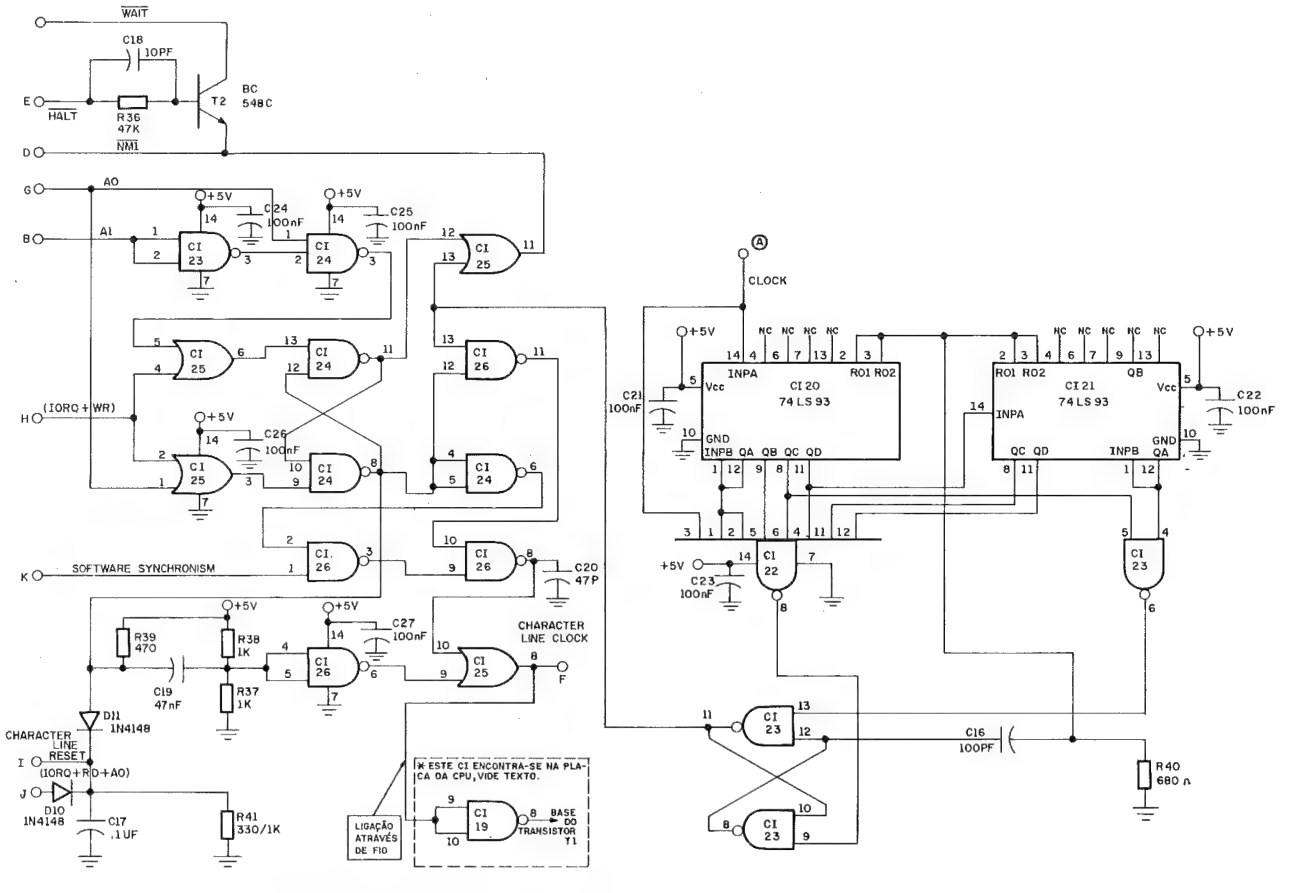


FIGURA 1 – CIRCUITO PARA IMPLEMENTAÇÃO DA FUNÇÃO SLOW

IMPLEMENTAÇÃO DO CIRCUITO

Não aconselhamos a implementação deste circuito a pessoas que não tenham larga experiência em montagem de circuitos eletrônicos digitais.

O circuito da função SLOW deverá ser disposto e montado numa placa de circuito impresso de 80 x 85 cm, conforme a figura 2, de modo que possa ser instalada através de quatro (04) parafusos com espaçadores isolantes, sobre a placa de CPU do NEZ-8000, no interior de sua caixa.

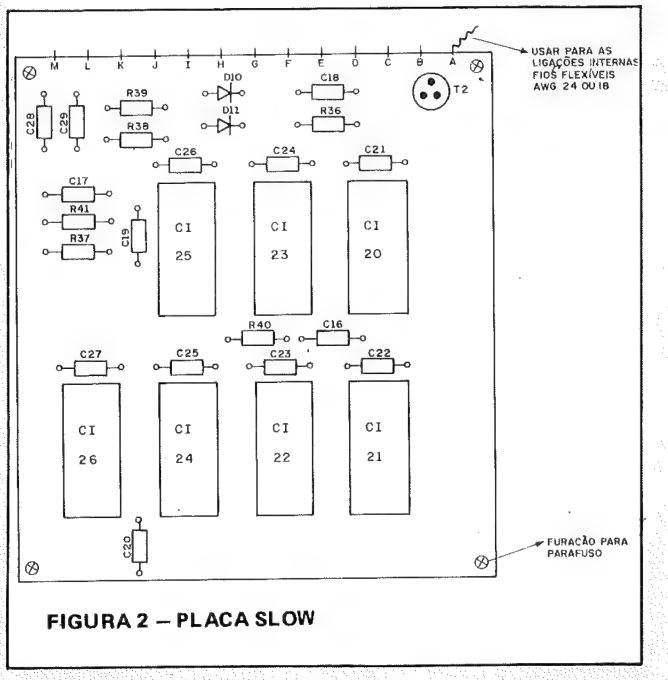


FIGURA 2 – PLACA SLOW

As ligações externas da placa SLOW deverão ser feitas utilizando-se fio flexível AWG 24 ou 18. Para sua melhor orientação, a figura 3 apresenta o layout da placa do NEZ-8000.

LIGAÇÕES A SEREM FEITAS

- O ponto "A" da placa SLOW deverá ser ligado ao pino 12 do CI5 (74LS05) do NEZ-8000.
- Corte a ligação do pino 6 do CI13 (74LS74) do NEZ-8000 com a base do transistor T1 (BC 237).
- Ligue o ponto "F" da placa SLOW para os pinos 9 e 10 (vide figura 1) do CI19 (74LS00) do NEZ-8000.
- Ligue a saída, pino 8 do CI19, à base do transistor T1 no NEZ-8000.
- Ligue o ponto "F" da placa SLOW para o pino 1 do CI14 (74LS93) do NEZ-8000.
- Corte a ligação do pino 1 do CI14 para o nó formado pelas junções das ligações do pino 5 do CI13 (74LS74), do pino 1 do CI11 (74LS74) e do resistor R 32, no NEZ-8000.
- O ponto "B" da placa SLOW deverá ser ligado à entrada A1 (conector de saída) ou ao pino 31 do Z-80.
- O ponto "G" da placa SLOW deverá ser ligado ao pino 13 do CI8 (74LS32) do NEZ-8000.
- Corte a ligação do pino 11 do CI9 (74LS00) para os pinos 2/3 do CI14 (74LS93) no NEZ-8000.
- Ligue o ponto "I" da placa SLOW ao pino 2/3 do CI14 (74LS93) do NEZ-8000.
- Ligue o ponto "J" da placa SLOW ao pino 11 do CI9 (74LS00) do NEZ-8000.
- Ligue o ponto "H" da placa SLOW ao pino 9 do CI9 (74LS00) do NEZ-8000.
- Ligue o ponto "K" da placa SLOW ao pino 5 do CI13 (74LS74) do NEZ-8000.
- Ligue o ponto "L" da placa SLOW a Vcc (+ 5 Volts), pino 3 do CI1 (7805) do NEZ-8000.
- Ligue o ponto "M" da placa de SLOW à terra, pino 2 do CI1 (7805) do NEZ-8000.
- Ligue o ponto "C" da placa SLOW à entrada $\overline{\text{WAIT}}$ (conector de saída, pino 19A) ou ao pino 24 do Z-80, no NEZ-8000.
- Ligue o ponto "D" da placa SLOW à entrada NM1 (conector de saída, pino 12A) ou ao pino 17 do Z-80, no NEZ-8000.
- Finalmente, ligue o ponto "E" da placa SLOW à entrada HALT (conector de saída, pino 13A) ou ao pino 18 do Z-80, no NEZ-8000.

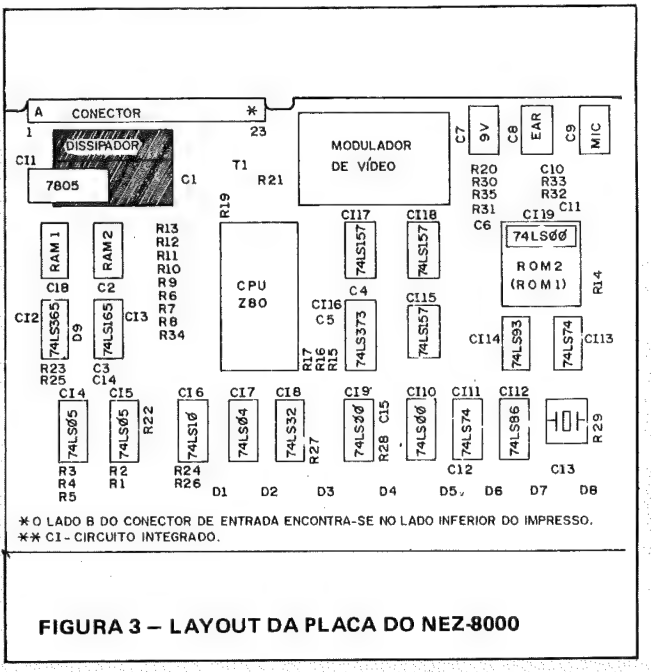


FIGURA 3 – LAYOUT DA PLACA DO NEZ-8000

Confira as ligações e teste o seu circuito. Digitando qualquer tecla a televisão não perderá o sincronismo horizontal.

Lembramos aos leitores que a implementação de circuitos digitais requer certas técnicas que não são ensinadas nas escolas tradicionais e nem tampouco nos livros (vide INTERFACE número 2, páginas 29, 30 e 31). Monte o circuito em *wire wrap* ou no *protoboard* antes de confeccionar a placa de circuito impresso. Teste o circuito, depure-o, tenha certeza que está funcionando como o previsto e depois confeccione o impresso.

Se você é estudante do curso técnico ou de engenharia, leve o projeto para sua escola, forme um grupo de pesquisa e estude o funcionamento de cada parte do circuito. Compre um esquema do hardware do NEZ-8000, veja as modificações introduzidas pela implementação do SLOW e analise a operação do circuito. No final, a experiência adquirida irá lhe proporcionar um *know-how* que só o dia-a-dia numa indústria de projetos permitiria ao profissional de engenharia ou ao técnico.

Acreditando na criatividade do brasileiro, convidamos aqui os "eletroniqueiros" a nos enviar os seus projetos ou inventos para publicação, onde mencionaremos nome e endereço, permitindo a troca de experiências entre nossos 30.000 leitores.

ASSEMBLER NO TK-82C

Por Mauricio Costa Reis

O microcomputador TK82-C utiliza um microprocessador Z-80 que possui um conjunto de instruções bastante poderoso, como pode ser visto no curso do Z-80.

Conhecendo cada instrução (ver manual do TK82-C, apêndice A) e mais algumas "dicas" (ver manual do TK82-C, cap. 25 a 28) já seria suficiente para utilizar a linguagem Assembler no TK82-C; no entanto, as informações estão muito dispersas, o que acaba gerando confusão no iniciante.

Neste artigo é apresentado objetivamente como usar a linguagem Assembler no TK82-C e assim conseguir maior rapidez na execução de programas, além de poder dispor de mais recursos que o BASIC fornece.

PROGRAMANDO

Um montador Assembler é um programa especial que permite a programação Assembler através de mnemônicos das instruções. Vários micros já dispõem de montador Assembler próprio, o que não acontece com o TK82-C, cujo montador Assembler é comercializado separadamente.

Então, como programar em Assembler no TK82-C com os recursos disponíveis?

Primeiro vamos ver um exemplo escrito em BASIC:

```
10  PRINT "*";
20  GOTO 10
```

Digite este programa e então execute-o em modo SLOW. Observe que após o RUN demora cerca de 10 s. para a tela ficar completa de asteriscos e aparecer o código de terminação 5/10 (tela completa provocada pela linha do programa BASIC).

O quadro seguinte mostra um programa Assembler que executa a mesma função do programa BASIC (preenche a tela com asteriscos):

RÓTULO	MNEMÔNICO	HEXADECIMAL	IMPRESSÃO
início	1d A, * call print jr início	3E 17 CD 08 08 18 F9	Y * LN gA gA / RAND

Vamos ver, então, o que faz este programa: *1d A, **carrega o registrador A com o valor do caractere * (17h); *call print* chama a rotina da ROM que imprime na tela o valor do registrador A; *jr início* volta ao início do programa para a impressão de novo asterisco. Como *jr* é um salto relativo, antes devemos saber quantos bytes compõem as instruções anteriores e então colocar o valor adequado (F9h significa um salto de 7 bytes, contando com zero a instrução seguinte ao F9h).

Existem dois modos de armazenar este programa. O primeiro é colocar o programa dentro de um comentário. Apague o programa anterior e digite:

```
10  REM Y * LN gA gA / RAND
20  RAND USR 16514
```

Onde *gA* é o símbolo gráfico da tecla A (obtido com GRAPHIC SHIFT A). Para digitar RAND digite antes THEN RAND e então apague o THEN. Isto é necessário pois o RAND só aparece quando o cursor está no modo key (inverso K). USR 16514 (o RAND é necessário devido a sintaxe do BASIC) executa uma rotina em Assembler a partir do endereço 16514, que é o endereço do primeiro byte após o REM.

Agora digite RUN newline e veja a rapidez que a tela é preenchida (cerca de 1 s.), mesmo em modo SLOW!

Substitua a linha 20 por:

```
20  PRINT AT 10,15; USR 16514
```

E veja a diferença. Isto acontece porque antes de imprimir o primeiro asterisco, o PRINT AT posiciona a tela na linha 10 coluna 15.

Pode acontecer que o código desejado não possua um caractere correspondente que possa ser impresso. Neste caso, a solução é reservar o espaço para o byte, dar um POKE adequado e, então, no comentário aparecerá um ?, indicando que o caractere é "não imprimível"!

Esta forma não permite muita flexibilidade para a confecção de programas em Assembler. A seguir é apresentado outro método para programação que facilita bastante o teste de programas.

O método consiste em escrever um programa Assembler no final da memória. Para isto é necessário "proteger" a parte final da memória do sistema BASIC (alterar o valor da variável RTP — Ram ToP). Digite então:

```
POKE 16388,80
POKE 16389,70
NEW
```

O valor de RTP no TK82-C com 2 Kb de memória é 18432. Com estes comandos o valor de RTP é alterado para 18000, e com isto restam 432 bytes para programas Assembler.

Só resta fazer um programa que facilite a entrada de programas escritos em Assembler. Digite o seguinte programa:

```
 10  CLS
 20  LET X = 18000
 30  LET A$ = " "
 40  IF A$ = " " THEN INPUT A$
 50  IF A$ = "P" THEN STOP
 60  IF A$ = "N" THEN GOTO 10
 70  PRINT A$ (2 TO) ; "." ;
 80  POKE X,16 * CODE A$ + CODE A$ (2) – 476
 90  LET X = X + 1
100  LET A$ = A$ (3 TO)
110  GOTO 40
```

Este programa permite a entrada direta de programas Assembler em hexadecimal, que será colocado então a par-

tir do endereço 18000 (4650h). À medida que os dados forem sendo digitados irão sendo impressos para verificação. Se houver algum erro digite "N" (não) e recomece tudo novamente. Para terminar, digite "P" (parar).

Os dados (o programa Assembler) estão na coluna HEXADECIMAL do quadro visto no início. Assim sendo, execute o programa (RUN newline) e irão aparecer duas aspas, indicando que o programa está pronto para a entrada de dados. Digite então:

3E17 newline CD0808 newline 18F9 newline

Observe a tela. Se estiver tudo OK digite "P"; caso contrário digite "N".

Pronto! O programa Assembler está armazenado na memória a partir da posição 18000. Agora vamos executá-lo! Digite:

RAND USR 18000 ou PRINT AT 10,20; USR 18000

Se você possui uma expansão de memória de 16 Kb use:

```
POKE 16388,48
POKE 16389,117
NEW
```

Substitua a linha 20 do programa por:

20 LET X = 30000

Para executar o programa Assembler digite:

RAND USR 30000 ou PRINT AT 10,20; USR 30000

Este segundo método não permite que o programa seja salvo pelo comando SAVE para a fite cassete. Para isto, digite NEW e então entre com o seguinte programa:

```
 1  REM 1234567
10  LET A = 16514
20  LET B = 18000 (30000 expansão de 16 Kb)
30  FOR I = 0 TO 6
40  POKE A + I, PEEK (B + I)
50  NEXT I
```

Repare que: o comentário (REM) deve possuir tantos caracteres quantos forem os bytes que formam o programa Assembler; *A* contém o endereço do primeiro byte após o REM; *B* contém o endereço do programa Assembler. Agora liste o programa e veja a surpresa! O comentário foi alterado para a mesma forma apresentada no primeiro método, podendo assim ser salvo pelo comando SAVE.

OUTRO EXEMPLO

O programa a seguir permite que o seu TK82-C emita sons através de sua TV! Parece brincadeira, não é? Experimente-o e veja os resultados: ao apertar uma tecla, a sua TV emitirá uma nota musical. O programa deve ser executado em modo FAST e não se esqueça de colocar volume na sua TV.

Rótulo	Mnemônico	Hexadecimal			Impressão		
notas	defm 9b,89,73	9B	89	73	i.	gG	115 1
	defm 69,00,93	69	00	93	105 1	space	i <
	defm 7E,00,5E	7E	00	5E	126 1	space	94 1
	defm 00,3B,31	00	3B	31	space	V	L
	defm 28,24,00	28	24	00	C	8	space
	defm 00,36,2C	00	36	2C	space	Q	G
	defm 00,00,00	00	00	00	space	space	space
	defm 0F,16,1E	0F	16	1E	?	—	2
	defm 00,0A,0C	00	0A	0C	space	gS	libra
	defm 12,1A,00	12	1A	00	>	,	space
	defm 00,00,41	00	00	41	space	space	INKEY
	defm 4C,00,38	4C	00	38	76 1	space	S
	defm 3C,46,53	3C	46	53	W	70 1	83 1
pausa	ld A,B	78			120 1	7	
volta	dec A	3D			X		
	jr nz, volta	20	FD		4	CLEAR	
	ret	C9			TAN		
início	call kscan	CD	BB	02	LN	iV	g^2
	ld B,H	44			68 1		
	ld C,L	4D			77 1		
	ld D,C	51			81 1		
	inc D	14			:		
	jr nz, início	28	F7		C	RUN 3	
	call findchr	CD	BD	07	LN	iX	gE
	ld DE, notas-126	11	D2	45	)	g4 2	RND 2
	add HL,DE	19			;		
	ld B, (HL)	46			70 1		
	xor A	AF			iJ		
	cp 8	B8			iS		
	jr z, início	28	EB		C	FOR 3	
	in A, (FF)	DB	FF		< =	COPY 3	
	call pausa	CD	77	46	LN	iD 2	i4 2
	out (FF),A	D3	FF		PEEK	COPY 3	
	call pausa	CD	77	46	LN	iD 2	i4 2
	jr início	18	E0		/	STEP	

Obs.:
- Não imprimível. Deve ser dado um POKE adequado quando usado em REM.
- O endereço é diferente quando a rotina está em REM.
- Digitar THEN antes (depois apagar o THEN).
- *g* significa símbolo gráfico da tecla correspondente e *i* significa caractere inverso (GRAPHIC caractere).

Agora digite:

FAST
RAND USR 18044

E pronto, agora você tem um órgão a seu dispor! As teclas 1, 2, 5, 9, 0, Q, P, A, F, K, L, newline, shift e space não produzem som.

CONCLUSÃO

Foram mostradas duas formas de programar em Assembler no TK82-C sem o uso do montador Assembler, o que apesar de trabalhoso possui muitas vantagens sobre o BASIC. O formulário de programação proposto facilita a programação nos dois métodos, apesar de ser mais adequado para o segundo método, que apesar de mais flexível, o programa criado por ele não pode ser salvo diretamente pelo comando SAVE do BASIC.

Dentro dos programas Assembler (tratados pelo BASIC como uma rotina USR e por isso deve terminar por um RET do Assembler) podem ser chamadas rotinas da ROM do TK82-C, facilitando muito a programação.

Mas não se esqueça: se o programa Assembler contiver um erro, o seu micro poderá "sair do ar" (apagar a tela e o teclado ficar sem função), quando então só restará desligar a fonte e começar tudo novamente.

Mas não desista, pois vale a pena!

BIBLIOGRAFIA

BAKER, Tony — Mastering Machine Code on your ZX81, Reston Publishing Company, Virginia (EUA), 1982.

1.º ENCONTRO REGIONAL DOS USUÁRIOS DE MICROCOMPUTADORES

(Promoção revista INTERFACE)

No mês de setembro próximo passado a revista INTERFACE reuniu durante quatro sábados, no Clube de Engenharia do Rio de Janeiro, a nova comunidade de usuários de microcomputadores carioca. Mantendo a tradição de não levar os seus problemas a sério, a presença dos usuários cariocas foi pequena, porém significativa. Neste encontro, levantou-se os principais problemas enfrentados pelos usuários.

Na mesa de debates destacou-se a presença do Engº Pierre Jean Lavelle, do Projeto Ciranda da Embratel, do Engº Ramiro de Araujo Almeida Sobrinho, do Grupo de Micros da Telerj, e dos representantes das lojas *Clappy Computadores, Microshow, Tecnocoop, Sistema, BCD Engenharia, Yatec, LHM, Microkit, Monk* (São Paulo), *Micron* (São Paulo) e *Jornal Microbits* (São Paulo), que num ambiente descontraído apresentaram suas experiências e opiniões sobre o mercado de microcomputadores.

"O usuário ainda é imaturo, sente-se inseguro para reivindicar os seus direitos e ainda não aprendeu a exigir do fabricante e do fornecedor toda a assistência necessária. As experiências dos grupos nos mostraram que a união faz a força, pois dentro de um grupo de micros troca-se experiências tanto de hardware como de software, e tem-se, também, o poder de pressão sobre o fabricante".

Como havíamos citado no editorial da revista número 11, o mercado regional de microcomputadores pode ser visto sob três aspectos diferentes: político-social, técnico e comercial.

Do ponto de vista técnico não discutimos a capacidade das empresas ou das universidades, mas sim uma maneira dessas duas se unirem em prol da tecnologia nacional; ou melhor, habituarem-se a caminhar juntas. Não vamos inventar a roda, é o caso do CP/M, porém, vamos criar um sistema operacional dentro da realidade nacional, possível de ser utilizado por qualquer microcomputador disponível no mercado e que possibilite ao usuário uma maior utilização das linguagens disponíveis, a um preço razoável. Alguns fabricantes nacionais fazem questão de desenvolver sistemas operacionais herméticos a um preço inacessível para o usuário. Devemos lembrar que o CP/M nos Estados Unidos não foi desenvolvido por nenhuma potência, no entanto até a IBM preparou o seu PC (Personal Computer) para rodar o CP/M. Tudo isto em respeito ao usuário, que pode desfrutar de uma variedade muito grande de programas disponíveis no mercado sem ter que investir altas quantias em desenvolvimento de software. Este é só um primeiro bate-papo, pois na próxima edição continuaremos.

Apresentamos nesta edição a palestra que foi apresentada no dia 17 de setembro pelo Dr. Tarcisio Neves da Cunha, mestre em Engenharia biomédica e Analista de sistemas. O tema apresentado foi a *Seleção de Software Aplicativo.*

Seleção de Software Aplicativo

Por Tarcisio Neves da Cunha

O *software* constitui um elemento singular nos sistemas de computação: excelentes máquinas com *software* deficiente perdem mercado para máquinas bem inferiores apoiadas por *software* poderoso. Ninguém ignora que o que vende é a aplicação e não a máquina.

No entanto, a produção de programas aplicativos exige criatividade e grande preparo da mão-de-obra especializada, associando imaginação com técnica. Exige também um esforço intenso em quantidade e qualidade, o que é o mesmo que dizer *custo.* É precisamente por isso que se diz que o *software* é o "gargalo" do desenvolvimento de sistemas: tanto no fator *tempo* como no fator *custo.*

Diante deste imperativo, prolifera-se a chamada *pirataria de software,* que provoca a desvalorização da mão-de-obra especializada aviltando os preços de mercado, que por sua vez realimentam a prática antiética. Para o usuário o resultado é o efêmero benefício do preço mais acessível, seguido de perto por um serviço mal prestado, manuais mal traduzidos, descontinuidade da assistência técnica e de atualizações posteriores, ausência de garantia etc, conseqüências estas da falta de absorção tecnológica por parte dos fornecedores.

Tratamos de examinar alguns pontos importantes na seleção de *software* aplicativo.

O AMIGO CHAMADO BOM SENSO

A necessidade conduz à aquisição. Só que por caminhos os mais diversos. Num dos extremos está o usuário que amadureceu uma idéia e procura com segurança o aplicativo que solucionará seu problema. No outro extremo está aquele cidadão que ao ver, por acaso, um aplicativo rodando, descobre que necessita muito dele. A grande diferença entre ambos é que o primeiro está protegido pelo *bom senso* ao contrário do segundo, cujo deslumbramento pode levar a uma precipitada compra de um "pacote de dores de cabeça".

O bom senso nascido do amadurecimento da idéia constitui 99% dos requisitos para uma boa decisão. Os restantes 1% podem ser creditados à sorte. Significa que só o usuário conhece perfeitamente sua necessidade. Logo, o que pretendemos aqui é fornecer pistas para reduzir seu risco.

COMPRAR OU DESENVOLVER: QUESTÃO DE ADAPTAÇÃO

A possibilidade de desenvolver seu próprio aplicativo não deve ser descartada logo de início, como também não deve ser adotada inflexivelmente.

De certo, existem aplicações mais que padronizadas cujo preço de venda torna evidentemente proibitiva sua reinvenção. Isto ocorre com a maioria dos pacotes de larga difusão no mercado, especialmente os que visam atender tarefas corriqueiras: contabilidade, pagamento, estoque, contas a pagar e receber, etc.

No entanto, há sempre uma diferença entre o que o usuário espera do pacote e o que este efetivamente fornecerá. A aplicação real sempre traz umas surpresas traiçoeiras, não previstas ou deliberadamente excluídas da faixa de atuação do aplicativo. E isto é perfeitamente normal. Cabe ao usuário detectar estas anomalias em potencial, e decidir: ou adapta o *software* às suas necessidades ou modifica seus métodos para que se enquadrem no programa.

Esta última alternativa não precisa ser necessariamente absurda. Existem "vícios" adquiridos ao longo do tempo que imperceptivelmente vão prejudicando o bom andamento de uma empresa, e só são isolados e erradicados graças à adaptação aos métodos computacionais sadios. Também em empresas de porte gigante, temos tido exemplos de grandes reformas administrativas objetivando a adaptação a métodos gerenciais assistidos por computador. É o caso dos grandes bancos de dados. Se eles funcionam bem, por que não segui-los?

De qualquer maneira são três os caminhos neste nível, todos válidos:

- *Desenvolver a aplicação de acordo com suas necessidades.*
- *Adaptar um pacote pronto.*
- *Adaptar-se a ele.*

O PACOTE E A MÁQUINA

Programas são escritos em formas rígidas chamadas *Linguagens de Programação.* Cada máquina está preparada para aceitar uma ou mais linguagens, daí a importância da *Compatibilidade* entre o pacote e a máquina específica de propriedade de cada usuário. Note-se que alguns microcomputadores são oferecidos no mercado com diversas linguagens, adquiríveis separadamente, o que implica que não basta ver um pacote rodando numa máquina igualzinha a que você tem em casa. Precisa ver se a linguagem usada está disponível também na sua máquina.

Além do mais, quando se trata de programa escrito em *assembler,* é importante estar atento ao fato de que *Cada Processador possui seu próprio Assembler.* Significa que programas escritos em *assembler* Z80 (por exemplo, o DGT 100, TK 82, etc) não rodam em 6502 (por exemplo o UNITRON), e vice-versa. Por este motivo a maioria esmagadora dos aplicativos é escrita em linguagem de alto nível (BASIC, por exemplo).

Ainda assim, existe um problema adicional relacionado com *Sistemas Operacionais* (por exemplo, o CP/M, DOS, UNIX, etc). Certos pacotes são desenvolvidos para rodar debaixo de um sistema operacional pré-determinado, o que exige que seus adquirentes possuam tal sistema.

Um parâmetro de máquina importante é a *quantidade de memória mínima* necessária para aguentar a carga do programa. Certos pacotes têm comportamento estável em

relação à memória disponível, ou seja, uma vez satisfeita a exigência de capacidade mínima, o programa roda com a mesma performance para quaisquer configurações de maior porte.

Mas às vezes tal não acontece. Algumas versões do *Micro Mumps* e do *Visicalc,* por exemplo, funcionam mal em máquinas com memória que apenas satisfaça o mínimo exigido. Ao se elevar um pouco a memória disponível, nota-se sensível melhoria de performance. Só que às vezes uma expansão excessiva volta a prejudicar o desempenho.

Tudo isso vale também para os discos magnéticos, quando exigidos pelos aplicativos, com alguns detalhes adicionais: deve-se ter atenção no tamanho do disco (8 ou 5,25 polegadas), na densidade de gravação (simples ou dupla), no uso ou não da segunda face e no tipo de formatação usado (sobretudo nos discos de 5,25").

De resto, considere os demais periféricos exigidos pelos diferentes pacotes. Em aplicações profissionais evite o K7, dada a sua baixa confiabilidade.

O ideal é que você consiga uma demonstração na sua própria máquina, e se isto ocorrer prepare-se: lá vão algumas pistas a mais.

NÍVEL DE SATISFAÇÃO

Nada como observar um pacote rodando na sua própria máquina, com seus próprios dados, operado por você mesmo. Aí é hora de você medir cuidadosamente certos parâmetros, sobretudo se houver possibilidade de escolha entre diversos fornecedores do mesmo aplicativo. Vamos lá:

- *Operacionalidade*

Quando mais fácil de operar um pacote, menor a probabilidade de erros. Comandos complicados ou ininteligíveis chamam erros, e podem destruir arquivos importantes. Por outro lado, pacotes mais complexos exigem operação mais complexa, mas não obrigatoriamente difícil.

- *Documentação*

É a base da sua segurança futura ao encarar o dia-a-dia da aplicação. Não precisa necessariamente de didática nem de apresentação (forma) impecável, mas é absolutamente imprescindível que seja *clara* e *completa,* de fácil acesso.

- *Resultado*

É o principal item: você precisa avaliar se os resultados apresentados espelham suas necessidades reais com a devida *objetividade.* Cerfitique-se da *precisão* e *confiabilidade* desejadas.

- *Tempo de resposta*

É o tempo decorrido entre o início e o final do processo para uma determinada base de dados. A importância deste parâmetro relaciona-se diretamente ao nível de satisfação desejado e serve para ajudar na seleção de pacotes compatíveis, considerado o custo de cada um. Se associarmos o tempo de resposta à performance, o *quociente preço/performance* dá uma avaliação quantitativa muito útil para a decisão final.

- *Benefício*

Outra relação quantitativa útil é o *custo/benefício,* mas nem sempre é fácil associar valores ao benefício advindo da aquisição e uso de um aplicativo. A economia de mão-de-obra de modo geral não se presta para avaliação de benefício já que guarda envolvimentos complexos. É preferível enveredar por caminhos do tipo competitividade, segurança, expansibilidade a custo zero etc.

- *Condições de contorno*

É uma avaliação feita rodando o pacote sobre uma base de dados criteriosamente escolhida de modo a explorar as situações mais adversas possíveis, a fim de validar o comportamento do programa. Por exemplo, num pacote de pagamento testar as mais diversas formas de horas extras, rotatividade de mão-de-obra e encargos sociais sobrepostos, como afastamento pelo INAMPS de um empregado com mais de uma pensão alimentícia, muitos dependentes e direito à insalubridade. .

PRINCIPAIS ÁREAS

Incluímos esta pequena classificação para ajudar a organização das idéias e despertar os usuários mais desligados para as opções a seu alcance.

- *Software básico*

Fogem um pouco da área de aplicativos, mas têm seu interesse; são:

— *Sistemas operacionais* como CP/M, DOS, UNIX etc.

— *Linguagens de programação* como BASIC, ASSEMBLER, FORTRAN, COBOL, MUMPS, etc.

— *Desenvolvimentais* como DEBUGGER (DEPURADOR), COMPILADORES CRUZADOS (exemplo: ASSEMBLER Z80 para micros 6502).

- *Jogos*

Os mais diversos aplicativos para diversão e distração, em diversos níveis de complexidade como é o caso do XADREZ.

- *Educativos*

Que apóiam os processos de aprendizagem em diversas áreas de conhecimento. Nosso mercado é deficiente nesta importante área.

- *Administração*

De grande utilidade em pequenas e médias empresas, constitui um fator de padronização da gerência econômico-financeira, de pessoal e de estoques.

- *Tecnologia*

Atinge áreas especializadas como ANÁLISES CLÍNICAS, TOPOGRAFIA, CONSTRUÇÃO CIVIL, SIMULAÇÃO DE PROCESSOS, GRÁFICOS POR COMPUTADOR, ETC.

- *Automação de escritórios*

Os editores de texto, as agendas eletrônicas, malas diretas e fichários magnéticos são ferramentas inestimáveis em departamentos e escritórios.

CONCLUSÃO

Sirva este ensaio de estímulo ao usuário no exercício sério da seleção criteriosa de pacotes para aquisição. Quanto mais preparado está um usuário para analisar criticamente o *software* disponível, maiores chances terá de fazer a opção certa. Além de sua satisfação garantida por suas próprias convicções (e não pelas da propaganda mercadológica), estará dando preciosa contribuição à melhoria do nível técnico geral dos programas disponíveis na praça. Isto pelo simples fato de que usuários bem informados e preparados não compram maus produtos; usuários exigentes valorizam bons produtos; e o resto a gente deixa por conta da lei da oferta e da procura. Nisto consiste o maior poder de barganha do comprador de *software:* saber o que está comprando é valorizar seu investimento.

COMUNIDADE DE PROJETISTAS DE HARDWARE E SOFTWARE - COPHAS

Nos últimos anos, em decorrência da evolução tecnológica e de diretrizes políticas, cresceu o nº de projetistas na área de informática. Em termos tecnológicos, a integração em larga escala (LSI) de circuitos eletrônicos permitiu que um número maior de profissionais tivessem acesso ao hardware das máquinas digitais. Isto foi sentido, também, na área industrial, pois o microprocessador permitiu que soluções antes inviáveis economicamente fossem implementadas e popularizadas.

Outro fato relevante foi a política de reserva de mercado na área de informática, que gerou inúmeros empregos para profissionais de eletrônica e informática. E o mais importante é que estes empregos não eram somente para manutenção e montagem, mas fundamentalmente para o desenvolvimento de hardware e software. É claro que muitas empresas emularam vários modelos importados, mas para isso foi fundamental a competência de profissionais qualificados.

Enfim, apesar de alguns desvios, os objetivos foram alcançados e hoje temos um bom número de projetistas no mercado.

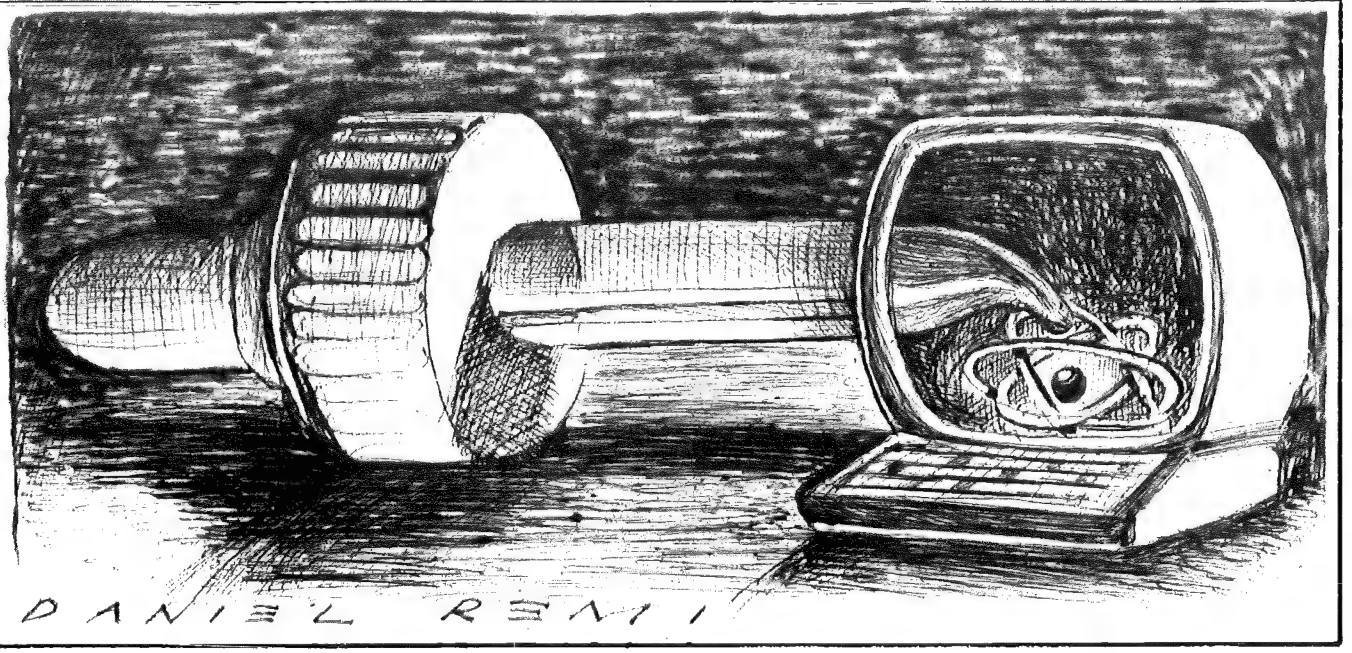
OBJETIVO

A nossa proposta é aproximar estes profissionais, através da revista INTERFACE, de tal modo que eles possam trocar idéias, prestar serviços, divulgar seus trabalhos e contribuir para o domínio tecnológico.

IDÉIAS

- O COPHAS terá um espaço fixo na revista INTERFACE em todas as edições.
- Será feito um cadastramento espontâneo destes profissionais interessados em associar-se à comunidade, incluindo:
  - Perfil do profissional.
  - Áreas de eficácia.
  - Sugestões.
- O COPHAS divulgará através da revista artigos técnicos e de projetos, incluindo todos os dados do autor para permitir que os interessados o localizem e procurem.
- Os associados utilizarão a revista para solicitar informações das quais eles não disponham, de tal modo que outro membro possa auxiliá-los e realizar alguma troca de idéias dentro do referido assunto.
- O COPHAS possibilitará:
  - Bolsa de emprego e oferecimento de serviços por parte dos membros da comunidade.
  - Banco de Dados com soluções já desenvolvidas, de tal modo que o membro da comunidade possa consultá-lo antes do projeto.
  - Benchmark de equipamentos nacionais.
  - Contribuição para a aproximação entre Universidade e Indústria.
- Aos melhores trabalhos enviados pelos membros da comunidade serão oferecidos mensalmente microcomputadores TK ou CP.

Inscreva-se e envie-nos seus trabalhos e sugestões.



GLOSSÁRIO DE TERMOS TÉCNICOS

BIT-SLICE PROCESSING
(Processamento bit-slice)

Processadores que permitem uma escala maior de processamento de dados em paralelo, isto é, permite que muitos trabalhos sejam realizados simultaneamente. Os processadores bit-sliced fabricados com a tecnologia bipolar, se comparados com os processadores CMOS, fornecem um sistema digital com muito mais versatilidade e uma velocidade de processamento 10 (dez) vezes maior. Os dispositivos bit-sliced diferem dos processadores convencionais principalmente na filosofia da arquitetura de suas CPU's. Enquanto os processadores convencionais têm o tamanho da palavra pré-definido (imutável) e um conjunto de instruções fixo, os processadores bit-sliced podem ser configurados por microprogramas, fornecendo uma ampla variedade de arquitetura com vários tamanhos de palavras e conjuntos de instruções. Os processadores bit-sliced são constituídos a partir de *slices* básicos de 4 ou 2 bits, que operam em paralelo e podem ser ligados em cascata fornecendo qualquer expansão que seja múltipla do *slice* básico. Desta forma pode-se construir microprocessadores de 8, 16, 32 e 64 bits de palavra (vide INTERFACE nº 5, 6, 7, 8 e 9).

BLOCK
(Bloco)

Um grupo de unidades de informação manipulado como uma única unidade. Em muitos dispositivos de armazenamento magnético somente blocos completos de informações podem ser acessados ou transferidos. O tamanho de um bloco pode ser fixo ou variável, dependendo do equipamento.

BLOCK DIAGRAM
(Diagrama em bloco)

Diagrama simplificado de um sistema eletrônico ou programa de computador, no qual as partes ou circuitos são representados por caixas interligadas (vide também diagrama de fluxo).

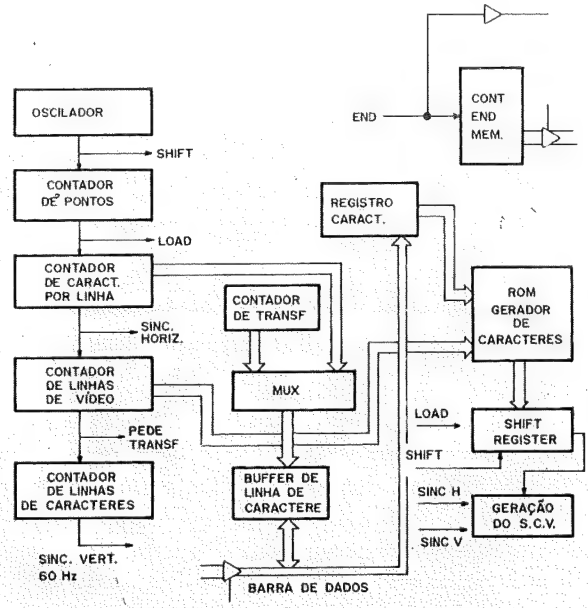


FIGURA – DIAGRAMA EM BLOCOS DO CONTROLADOR

BLOCK LENGTH
(Comprimento de bloco)

Tamanho ou comprimento de um bloco de registros, palavras, caracteres ou elementos binários.

BLOCK MARK
(Marca de bloco)

Um caractere especial que é utilizado para sinalizar o final de um bloco de dados, sobretudo quando se utilizam blocos de tamanho variável.

BLOCK TRANSFER
(Transferência de blocos)

Movimento de dados em blocos.

BLOWING

Termo utilizado em computação na referência a programação de memórias ROM (Read Only Memory) ou PROM (Programmable Read Only Memory), utilizando equipamento especial.

BMS™
(Business Management System)

Sistema de administração de negócios, marca registrada da BURROUGHS.

BOARD
(Placa)

Placa de circuito impresso de fibra ou fenolite (epoxi), na qual uma fina camada de metal é aplicada formando filetes que conduzem sinais elétricos e permitem a fixação, através de solda, dos componentes eletrônicos.

BOOLEAN ALGEBRA
(Álgebra booleana)

Álgebra utilizada na análise de informações e problemas em computadores, em virtude de expressar relações lógicas em formas que podem ser acomodadas dentro da lógica binária dos computadores digitais. Contrasta com a álgebra convencional que partilha apenas com relações matemáticas.

BOOLEAN OPERATION
(Operação booleana)

Qualquer operação em que cada um dos operandos e o resultado é um de dois valores. Tais operações podem claramente ser expressas em forma binária.

BOOTSTRAP

Versão primitiva de um programa de computação, usada para estabelecer outra versão. Rotina de iniciação que passa o controle do processo ao sistema operacional. Também conhecido como dispositivo ou técnica para carga de primeira instrução ou palavra de uma rotina na memória. Essas instruções são usadas para introduzir o restante da rotina. Normalmente, o *bootstrap* é um programa usado para inicializar o computador, limpando a memória, ativando os dispositivos e carregando o sistema operacional.

BOT

Início de fita (vide fita magnética).

BPI
(Bits por polegada)

Usado para medidas de densidade de dados em armazenamento.

BPS
(Bits por segundo)

Usado para indicar velocidade de transferência de informação.

BRANCH

Um ponto no programa onde o computador seleciona uma de duas saídas, ou mais.

BREADBOARD

Placa experimental de um circuito eletrônico para projeto e teste operacional.

BREAK POINT
(Ponto de interrupção)

Um intervalo no qual o programa ou rotina podem ser interrompidos mediante intervenção exterior.

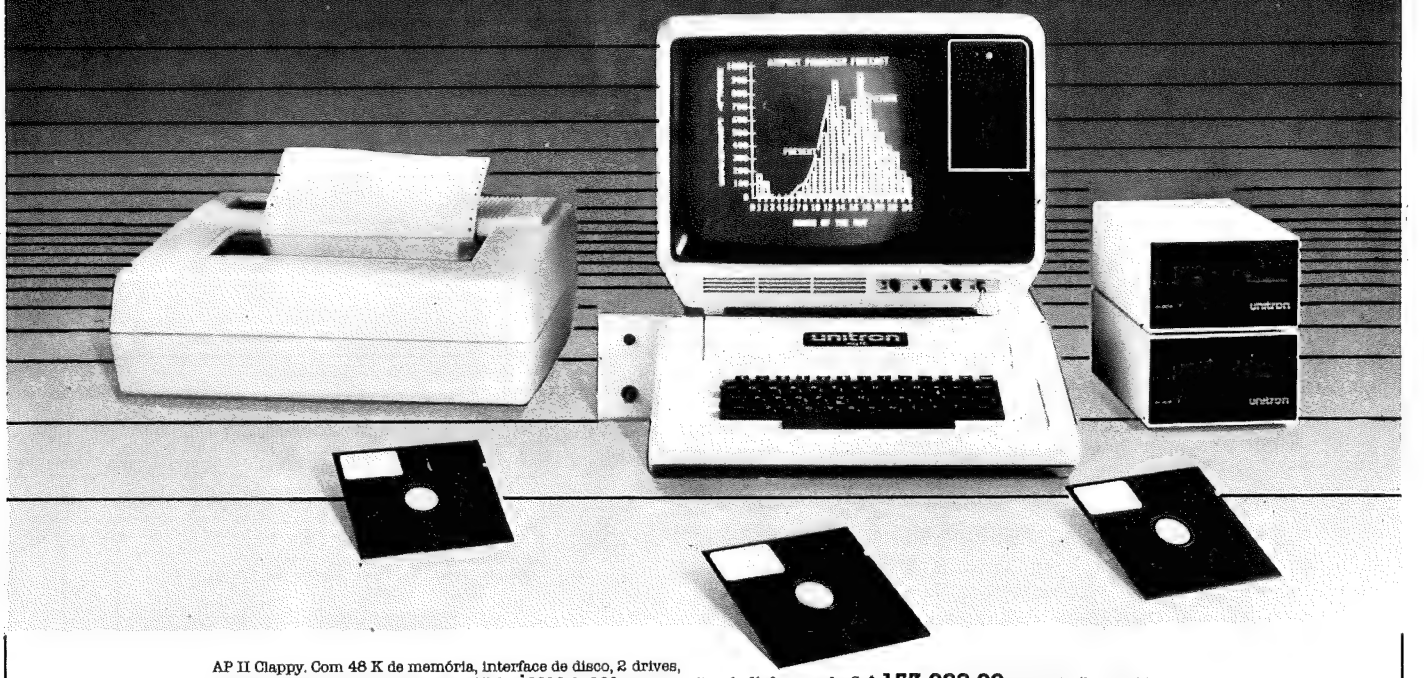
BROADCAST

Disseminação de informações para muitos receptores simultaneamente, na maioria das vezes via sinal eletromagnético.

BTAM
(Basic Terminal Acess Method)

Método básico de acesso a um terminal, utilizando canais de telecomunicações.

A Clappy apresenta o microcomputador com macrosoluções.



AP II Clappy. Com 48 K de memória, interface de disco, 2 drives, interface paralela, cabo, impressora Elebra 8010 de 100cps. e monitor de fósforo verde, Cr$ **153.000,00** mensais (leasing)*

É o AP II da Unitron. A solução perfeita para as pequenas, médias e grandes empresas, profissionais liberais e o dia-a-dia do lar.

Solução nas grandes empresas — O AP II pode ser utilizado no gerenciamento de departamentos, em orçamentos, PERT, previsões de vendas e no mais que você precisar ou prever.

Solução para o profissional liberal — O AP II da Unitron também foi criado sob medida para as necessidades de qualquer profissional liberal. No caso do médico e do advogado, por exemplo, ele faz o cadastro dos clientes, emissão de laudas, controle de medicamentos, controle de processos, cobranças judiciais e jurídicas.

Solução no escritório — O AP II faz tudo para minimizar e facilitar o seu trabalho: contabilidade, folha de pagamento, orçamentos, controle de estoque, contas a pagar e a receber com muito mais rapidez e eficiência.

Solução em condomínios — O AP II facilita ainda a vida no condomínio: na administração do orçamento, controle de cobranças e pagamentos, balancetes, folha de pagamento, controle de entrada e saída de visitantes, ficha de moradores.

Solução no preço — Na Clappy, você encontra o AP II pelo menor preço da praça e com macrosoluções de pagamento.

A maior loja de informática do Brasil
AS 100 MAIORES EDIÇÃO 83
REVISTA DADOS E IDEIAS

*Taxa Banco IOCHPE em 26.10.83.

unitron Clap Clappy

Venha à nossa loja ou solicite a visita de um representante.

Centro: Av. Rio Branco, 12 - loja e sobreloja.
Centro: Rua Sete de Setembro, 88 - loja Q (galeria).
Copacabana: Rua Pompeu Loureiro, 99.
São Cristóvão: Rua Antunes Maciel, 25 - 2.º andar.

Tels.: (021) 222-5517 • 234-0214 • 264-2096 • 253-3395 • 222-5721 • 267-4398 • 253-3170 • 236-7175 • 283-3588

Entregamos em todo o Brasil pelo reembolso Varig.

COPACABANA
Aberta diariamente das 10 às 20 horas e aos sábados das 10 às 15 horas.

FILIA

UM SISTEMA DE PROGRAMAÇÃO PARA MICROCOMPUTADORES

6ª PARTE

Por Rildo Pragana
Departamento de Física – UFPE – RECIFE

Nesta edição iniciaremos nosso artigo dando continuidade a apresentação da *extensibilidade de Filia em vários níveis* e abordando ainda a *implementação de um assembler com < COMP e EXEC >* e a *codificação dos metacompiladores.*

A primeira idéia é a de termos entrada e saída redirigida, sendo que cada dispositivo tem sua rotina (diferente) de leitura da entrada de 1 caractere, saída de um caractere, rotina para avançar uma linha, rotina para iniciar uma nova página e rotina para executar um *backspace*. Tais dispositivos podem ser um terminal, uma impressora ou um arquivo em disco por exemplo.

O nosso objetivo é escrever programas que possam ter suas saídas dirigidas para o terminal (ou tela do monitor se não existir um terminal independente, como no TRS-80, Digitus e similares, por exemplo), de modo que o mesmo programa para mostrar o conteúdo da memória possa servir para mostrar o conteúdo na impressora, no terminal ou em outra saída qualquer, e também, para que a entrada possa ser qualquer.

```
CAR: (define um vetor de entrada/saída. Na execução devolve
esse vetor numa variável chamada CARVETOR).
< COMP )*  ( passa para o modo de compilação )
EXEC > CARVETOR = ;        ( guarda o endereço do vetor
                           compilado  na variável CARVETOR )
```

A palavra)* serve para mudar o interpretador externo de Filia para o modo de compilação. Após sua execução (isto é, quando CAR: for executado) os átomos seguintes serão compilados ao invés de serem executados, daí formando o vetor de rotinas que desejamos, veja figura 1.

```
(CAR) ( fornece o deslocamento de um átomo de entrada/saída
no vetor )
   <COMP B, EXEC > B @CARVETOR @+ @ EXEC ;
```

O compilador (CAR) é usado para definir o deslocamento de cada função de entrada e saída no vetor. Durante a compilação é criada a função de E/S com o seu deslocamento e durante a execução o deslocamento dado à função correspondente será somado ao endereço de base do vetor (no caso CARVETOR) e a função armazenada no vetor será executada.

Após essa definição podemos entrar com os nomes das funções de entrada e saída redirecionáveis:

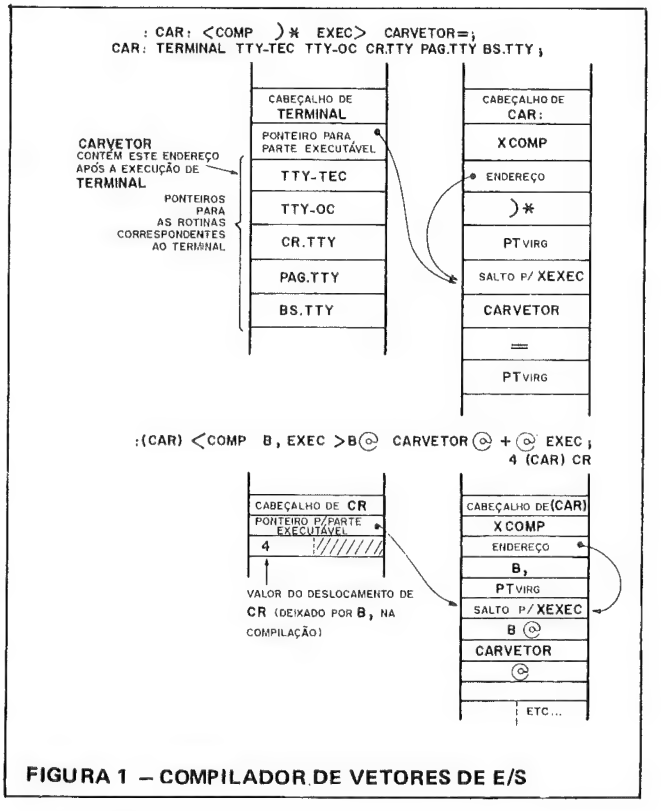


FIGURA 1 – COMPILADOR DE VETORES DE E/S

```
0 (CAR) TEC  (lê uma tecla)
2 (CAR) OC   (envia um caractere)
4 (CAR) CR   (avança uma linha)
6 (CAR) PAG  (avança uma página)
8 (CAR) BS   (backspace)
```

Observe que o deslocamento da função seguinte é 2+ o da anterior, porque cada vetor na verdade é o ende-

reço de uma rotina, que deve ser armazenado em 2 bytes.

Definiremos agora as rotinas de entrada e saída específicas de cada periférico:

```
CODE TTY-TEC (lê uma tecla do terminal)
      INÍCIO; USART-STATUS IN 2 ANI (testa o status 'receptor pronto')
        NZ/ ATÉ, SART-DADOS IN 7F ANI (retira o bit de paridade)
      A L MOV 0 H MVI (põe o caractere como 16 bits em HL)
      H PUSH PROX JMP (e devolve-o na pilha)
CODE TTY-OC (envia um caractere ao terminal)
      INÍCIO, USART-STATUS IN 4 ANI (testa status 'transmissor pronto')
        NZ/ ATÉ, H POP L A MOV USART-DADOS OUT PROX JMP
: CR . TTY (envia um retorno de carro para o terminal)
      OD TTY-OC 0A TTY-OC ; (envia 0DH seguido de 0AH, ou seja, cr-lf)
: PAG . TTY (avança uma página no terminal)
      CR . TTY CR . TTY CR . TTY CR . TTY ;
(esta rotina simula uma página no terminal, avançando 4 linhas)
: BS . TTY  8 TTY-OC ;
```

Agora as rotinas para saída numa impressora. Dependendo das características da interface com a impressora, algumas executam CR com apenas o 0DH, sendo o *linefeed* automaticamente gerado, algumas não possuem *backspace* etc . . .

As rotinas dadas servem para a Emília da Elebra.

```
CODE PRT-OC  (ENVIA um caractere para a impressora, conectada a outra
              Usart diferente, evidentemente, da do terminal de vídeo)
      INÍCIO, PUSART-STATUS IN 4 ANI NZ/ ATÉ, H POP L A MOV PUSART-DADOS
OUT PROX JMP (poderia ser incluído também um teste para o ^S ^Q do protocolo,
dependendo deste ser usado ou não).
: PRT-TEC ; (Esse programa não faz nada pois a impressora não envia "teclas"
            para o computador)
: PRPAG 0C PRT-OC ;
: PRCR 0D PRT-OC ;
: BSPRT ; (também não faz nada, pois não podemos "recuar" a impressora)
```

Eventualmente, outras rotinas que verificassem o final de uma folha de papel e efetuassem automaticamente o avanço de página poderiam ser incorporadas, no lugar das rotinas dadas, o que foge ao nosso objetivo atual. Finalmente, podemos definir as funções que redirigem a entrada e saída do sistema:

```
CAR: TERMINAL TTY-TEC TTY-OC CR . TTY PAG . TTY
BS . TTY ;
CAR: IMPRESSORA PRT-TEC PRT-OC PRCR PRPAG BSPRT ;
```

Executando então TERMINAL teremos toda a entrada e saída subseqüente do terminal. Isso porque colocaríamos em CARVETOR o endereço do vetor de rotinas TERMINAL definido acima e então a execução de cada (CAR) como TEC, OC, PAG etc . . . iriam executar a rotina correspondente ao terminal. Executando IMPRESSORA teremos a saída (e entrada) dirigida para a impressora. Evidentemente, podemos criar outras rotinas que façam entrada/saída na memória, disco, arquivos etc. O mecanismo dado é perfeitamente generalizado para que possamos tratar entrada e saída de uma maneira transparente para o usuário do sistema, que teria apenas que usar as funções definidas por (CAR), dadas acima.

Outra função interessante, que chamaremos de → , permite que façamos variáveis com um comportamento diferente do usual. Colocando → antes do nome da variável, teremos o topo da pilha armazenado nesta variável. Se, por outro lado, o nome da variável vier sem o → antes (algum instante antes) o valor da variável será devolvido, mesmo sem o uso do operador @. Teremos que redefinir variáveis (o compilador VAR) antes de serem criadas tais variáveis "especiais".

```
0 VAR  VAR-ACESSO (uma variável normal que servirá como flag)
: – > 1 VAR-ACESSO  = ; (essa função liga o VAR-ACESSO)
: VAR < COMP , EXEC > VAR-ACESSO @ SE  =
        0 VAR-ACESSO  =  SENÃO @ ENTÃO ;
```

Fazendo 25 VAR TESTE teremos em seguida:

```
      TESTE .  (e será impresso o valor de TESTE)
      TESTE 3 + → TESTE  (TESTE será somado de 3)
      TESTE DUP + → TESTE  (TESTE será multiplicado por 2)
```

O funcionamento de → é bastante simples: ele apenas liga o flag VAR-ACESSO. A compilação do novo VAR também é semelhante ao antigo.

A diferença é que se (durante a execução) o VAR-ACESSO se encontra ligado (presumivelmente pela execução de → anteriormente) será armazenado por ' =' o topo da pilha na variável e em seguida desligado o flag. Caso contrário, será executado um '@' para obter o valor da variável.

A vantagem desse método é que se evita o "esquecimento" de muitos programadores do uso do átomo "@ " para obter o valor de uma variável, e a confusão entre valor e endereço, que muitos fazem.

IMPLEMENTAÇÃO DE UM ASSEMBLER COM < COMP E EXEC >

Os metacompiladores estudados até agora podem ser muito úteis para implementar um assembler para o 8080/ Z80 ou mesmo outros microprocessadores.

Para isso devemos antes nos munir de uma tabela de *opcodes* do processador em binário (ou octal, hex etc . . .) para analisar as similaridades entre instruções que possuem em comum uma determinada operação. Exemplos disso são ADD B , ADD C , ADD D , etc, que diferem apenas em três bits. Estes 3 bits correspondem a um código interno do registrador usado como operando. Os saltos condicionais do 8080, por exemplo, são todos do tipo: (em binário).

1 1 x x x 0 1 0, onde x x x são bits que variam com as condições do salto. Classificando convenientemente as instruções do processador, poderemos então definir compiladores para *famílias de instruções.*

No assembler que mostraremos, uma grande quantidade de instruções são classificadas como instruções de um byte, num grupo à parte, que possuem em comum a imersão desse único byte no dicionário no instante de sua execução.

O compilador para essa família será:

```
: 1BYTE <COMP B, EXEC > B @ B , ;
```

podemos agora definir estas instruções:

```
00    1BYTE NOP     76 1BYTE HLT 0F3 1BYTE DI 0FB 1BYTE EI
07    1BYTE RLC     0F 1BYTE RRC  17 1BYTE RAL 1F 1BYTE RAR
0E9   1BYTE PCHL  0F9 1BYTE SPHL 0E3 1BYTE XTHL 0EB 1BYTE XCHG
027   1BYTE DAA     2F 1BYTE CMA  37 1BYTE STC 3F 1BYTE CMC
```

Existem instruções que, apesar de ocuparem um byte precisam de um operando *registrador* que deverá entrar num campo de 3 bits na instrução.

Como os três bits do registrador nestas instruções são os três menos significativos, basta somarmos ao código básico dessa instrução (com zeros nos 3 bits) um código para o registrador, que deverá ser introduzido na pilha antes de evocada a instrução. Esse é o motivo pelo qual o assembler em Filia também tem notação polonesa, com os registradores e outros operandos fornecidos antes da operação. Os códigos para os registradores foram vistos no artigo anterior. Compare o código produzido com o encontrado em alguma tabela de instruções do 8080 ou Z80.

```
: ARIT (compilador das instruções aritméticas lógicas)
      < COM B, EXEC > B @ +  B, ;
80    ARIT ADD   88 ARIT ADC  90 ARIT SUB   98
ARIT SBB
0A0   ARIT ANA 0A8 ARIT XRA 0B0 ARIT ORA  0B8
ARIT CMP
```

Posteriormente veremos o restante da implementação desse assembler. Discuta como seria escrito um assembler para o 6800 (sua solução é bem-vinda para a minha análise e correções), 6502 ou outros microprocessadores populares.

Para os mais gananciosos, sugiro escrever um assembler em Filia para o 8086, que é um processador de 16 bits. Observe que esse software não rodará no seu processador, devido à incompatibilidade do código objeto, mas poderá servir para transportar Filia para outros computadores. Na verdade, foi dessa forma que escrevemos Filia em vários micros diferentes.

CODIFICANDO OS METACOMPILADORES

Mostraremos agora como são escritos os compiladores < COMP EXEC > . Ambos produzem na memória chamadas (em Filia) para rotinas XCOMP e XEXEC, que são executadas pelo compilador quando definido. A função < COMP, em particular, produz como resultado a compilação de XCOMP, seguida de um espaço onde será posteriormente colocado o endereço após o EXEC > e deixa na pilha o seu endereço para que EXEC > possa colocar posteriormente seu endereço.

Reveja a figura 1 para observar como é a estrutura do código compilado de um compilador (no exemplo dado, CONST). O código dado usa SPUSH como macro para um *push* na segunda pilha (ponteiro SS).

```
CODE XCOMP  D  SPOP (retire DE da segunda pilha)
      HEAD  LHLD H INX H INX H INX H INX H INX H INX H INX
            (endereço do código –2 da última definição no dicionário)
            (esta definição corresponde ao novo membro da família)
      D LDAX A M MOV D INX H INX D LDAX A M MOV D INX
            (move endereço após XCOMP para o descritor do membro)
      D SPUSH (retorna o endereço de retorno, por sobre o número
            após XCOMP para a segunda pilha)
      PROX JMP
CODE XEXEC  B  SPUSH (salva PPC na pilha de retorno)
      D DCX (DE contém endereço indireto deixado por PROX)
            2 H LXI D DAD H PUSH (volta o endereço de parâmetro na pilha)
      XCHG M C MOV H INX M B MOV B INX B INX B INX
            (os três B INX são para evitar a interpretação da instrução de
            salto a XEXEC deixada no compilador)
      PROX JMP (agora será interpretada a parte comum à família do
            compilador)
:     YCOMP CODE XCOMP ; (o CODE cria o cabeçalho)
: <   COMP IMED 'YCOMP 2 – , (compila o endereço de YCOMP)
            AQUI 0 , ; (deixa o endereço na pilha e compila um
                  espaço para o EXEC posteriormente
                  escrever por cima)
: EXEC >  'PTVIRG 2– ,  (código meramente formal. Devemos conhecer o
                  endereço de PTVIRG e compilar como um número)
      AQUI TROQ =  (devolve o endereço seguinte para o espaço
                  deixado por COMP)
      'JMP B B ,  (código para o salto. Poderíamos ter colocado
                  0C3 no lugar de 'JMP B, mas para o 8080)
      'XEXEC 2– ,  (o endereço de salto é o XEXEC)
      IMED ;  (fim do programa, que deverá ser imediato)
```

O entendimento da implementação de < COMP EXEC > não é muito importante para um bom programador em Filia, mas a sua utilização sim. Portanto, não queremos desesperar os leitores que não tenham conseguido entendê-los completamente. Dúvidas e sugestões para implementação em outros micros são bem-vindas. O próximo mês nos trará alguns exemplos de programas em Filia, uma vez que agora vocês podem se considerar iniciados na programação desta nova linguagem. Mostraremos como é possível escrever funções geralmente difíceis de serem escritas em Basic ou outras linguagens similares, e o final do artigo (por volta da 9ª ou 10ª parte) terá como brinde final um glossário das funções de Filia e uma série de referências comentadas, onde os leitores poderão se especializar no uso dessa fascinante linguagem. ■

MICRO NEWS

Microcomputadores com crédito direto ou leasing

JR : Cr$ 176.000, x 3
CP - 200 : Cr$ 110.000, x 2
DGT - 1000: Cr$ 268.000, x 3
TK - 85 : Cr$ 115.000, x 2

GRÁTIS DE 10 À 20 JOGOS
ENTREGA RÁPIDA EM QUALQUER PARTE DO PAÍS E CURSOS BASIC, COM ATÉ 100% DE DESCONTO

Preços sujeitos a alteração

Sistemas Comerciais: contrôle de estoque; contabilidade; mala direta/editor texto; contas a receber/pagar; folha de pagamento e desenvolvimento de Software para cada necessidade. Toda linha de Suprimentos e Periféricos para acompanhar o crescimento de pequenas e médias empresas.

MICRONEWS COMÉRCIO E SERVIÇOS LTDA.
R. Assembléia 10 Gr. 3317 - Ed. Centro Cândido Mendes
Tel.: (021) 252-9420 - CEP 20011/RJ.

DESART

"BANANA 85"

O MICRO DA NOSSA TERRA

Microcomputador de baixo custo para desenvolvimento e aprendizado de software e hardware dos microprocessadores 8080/85/Z-80.

Características:

- RAM 1K → 5K.
- PROM 2K.
- Execução de programas instrução por instrução, a fim de facilitar depuração de programas mais complexos.
- Acesso direto à memória e registros.
- Entrada e saída paralela.
- Comunicação serial RS 232 C (V24) com velocidade de 75, 150, 300, 600, 1200, 4800 e 9600 selecionada por teclado.
- Sinal audível das teclas ativado pelo teclado.
- Leitura e gravação para cassete com circuito PLL e freqüências de 1200 Hz bit 0 e 2400 Hz bit 1, possibilitando a transferência de programas por telefone.
- Busca automática de programas no cassete.
- Conversão direta entre Decimal, Hexadecimal e Octal.
- "Set" dos caracteres ASCII, possibilitando testes rápidos de impressoras, terminais de vídeo, modems assíncronos, etc.

SUPORTE
ENGENHARIA DE SISTEMAS DIGITAIS

Rua Curuzu, nº 17 – São Cristóvão – RJ Telefone 580-7886

livros

RUBENS, André Gil e OLIVEIRA, Ney Acyr R. de — "MICROPROCESSADOR Z-80 (HARDWARE)", A e N — Consultoria Projetos e Publicações Ltda, Rio de Janeiro, 1983.

O presente livro é dirigido aos estudantes, profissionais e autodidatas das áreas de eletrônica e informática.

A filosofia que se segue no desenvolvimento do trabalho é a de permitir uma boa assimilação do assunto, tanto para os profissionais e estudantes que já conhecem algum outro microprocessador, como para os iniciantes nas áreas de eletrônica e informática. Para tanto, apresenta nos dois capítulos iniciais inúmeros conceitos indispensáveis ao entendimento dos microprocessadores. Acredita-se que o leitor que possua conhecimentos básicos de eletrônica digital e de álgebra booleana tenha mais facilidade para assimilar toda a plenitude do trabalho.

O leitor encontrará no decorrer dos capítulos uma enorme preocupação com os aspectos didáticos dos diversos conceitos envolvidos. Isto porque, uma vez dominado o hardware de um microprocessador como o Z-80, torna-se bem mais fácil compreender e utilizar outros microprocessadores, pois os conceitos básicos de funcionamento permanecem inalterados.

O capítulo 1 é eminentemente conceitual. Ele é muito importante para o leitor que está iniciando na área de informática, pois nele é feito um nivelamento dos conhecimentos básicos de computadores. Apresenta um histórico das máquinas de computação, dando ênfase aos fatos que influenciaram nas características das modernas máquinas de hoje existentes no mercado, em especial o microcomputador. Em seguida, apresenta a organização básica de um computador digital e sua seqüência operacional.

O capítulo 2 conceitua microprocessador e microcomputador. Apresenta a arquitetura básica, memórias, periféricos e as principais aplicações dos microcomputadores. Finaliza com um breve histórico dos microprocessadores, o que permite ao leitor situar-se em relação ao Z-80.

O capítulo 3 é dedicado ao estudo da arquitetura específica do Z-80, mostrando a sua organização lógica e o seu mapa de registradores.

No capítulo 4 são descritas as características elétricas de todos os sinais do microprocessador Z-80 e a sua compatibilidade com as famílias TTL e CMOS.

No capítulo 5 são apresentados os estados, os diversos ciclos de máquina e os diagramas de tempo de cada um destes ciclos do Z-80.

O capítulo 6 foi reservado exclusivamente para o estudo das interrupções no Z-80. Primeiramente, faz um estudo conceitual do assunto para, em seguida, apresentar separadamente os modos de interrupção do Z-80. Foi dedicado um capítulo a este assunto dada a sua importância nos sistemas em tempo real.

O capítulo 7 apresenta exemplos de projetos didáticos de módulos de memória e de interfaces de entrada e saída.

O capítulo 8 procura reunir no projeto de um microcomputador todos os conhecimentos adquiridos nos capítulos anteriores. Descreve, também, a interface paralela PIO-Z80, que faz parte do referido projeto.

No final de cada capítulo o livro apresenta diversos exercícios de fixação. Estes exercícios fazem parte integrante do aprendizado do texto, pois ao tentar resolvê-los o leitor irá consolidar e ampliar os conhecimentos adquiridos. Tem um total de cento e quarenta exercícios, incluindo desde perguntas objetivas sobre tópicos do texto até projetos de interfaces e módulos de memória.

COMO OBTER O LIVRO PELO CORREIO, EM SUA CASA:

NOME _____

ENDEREÇO _____

CID. _____ EST. _____ CEP. _____

PROF. _____ CARGO _____

DESEJO RECEBER PELO CORREIO
PARA ISSO ESTOU ENVIANDO CHEQUE NOMINAL Nº _____ BANCO _____
NO VALOR DE: Cr$ 6.800,00

_____
ASSINATURA

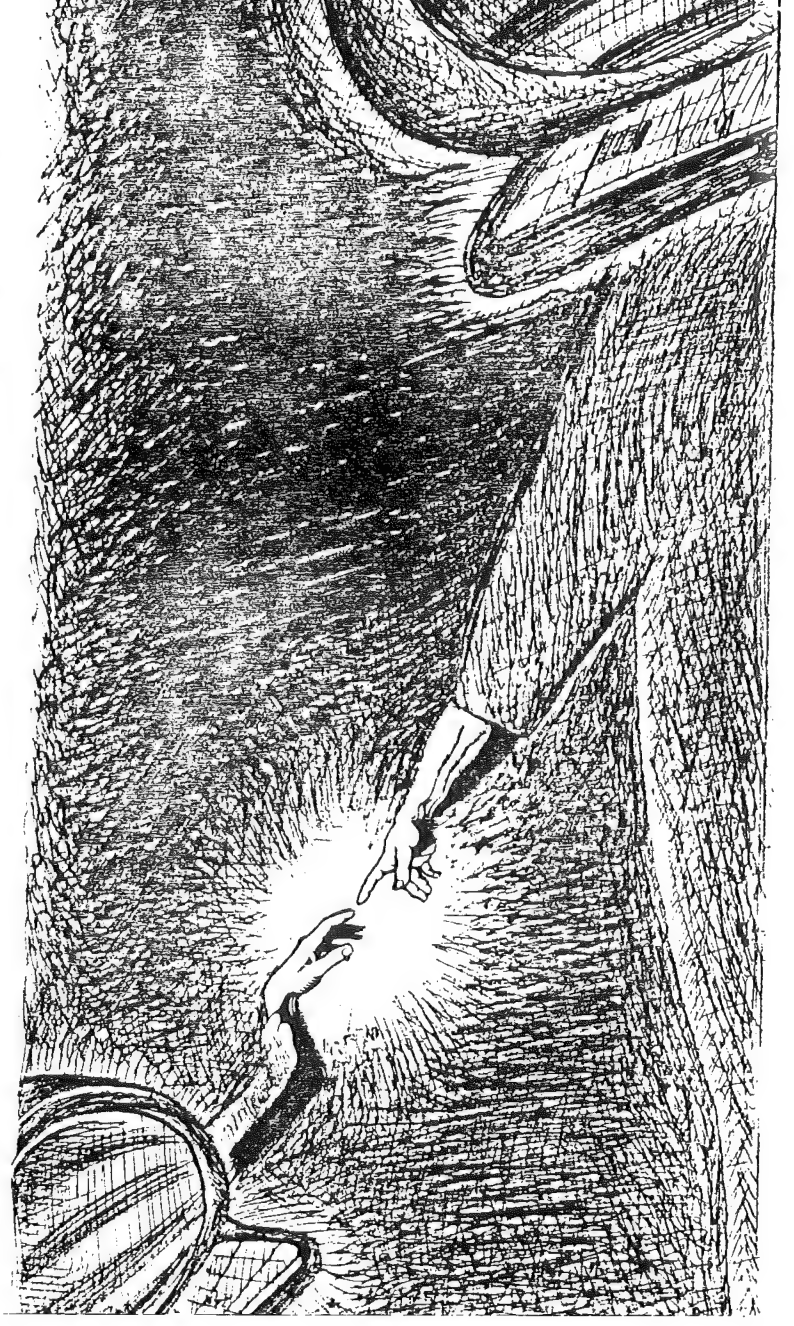
A e N — Consultoria Projetos e Publicações Ltda
Rua da Conceição, 105 — sala 1906 — Rio de Janeiro — RJ — CEP 20051

MUMPS

CARACTERÍSTICAS E COMPARAÇÕES COM OUTROS SISTEMAS DE PROGRAMAÇÃO

Por T. Munnecke. R. F. Walters, J.Bowie, C.B. Lazarus e D.A. Bridger

4ª Parte



MANIPULAÇÃO DE DADOS DBTG

O *Data Base Task Group* (DBTG) da CODASYL (*Conference on Data Systems Languages*, uma organização composta de usuários e fabricantes de computadores selecionados, e a criadora do COBOL) foi criado com o objetivo de definir uma padronização para a descrição e manipulação de banco de dados. O DBTG publicou um relatório final em abril de 1971 contendo uma proposta detalhada para linguagens de descrição, manipulação e controle físico de dados.

O DBTG subseqüentemente dissolveu-se (sua missão estava completa), mais dois novos grupos da CODASYL, chamados DBLTG (Data Base Languages Task Group) e DDLC (Data Definition Language Committee) foram formados para analisar e posteriormente desenvolver as partes COBOL e não-COBOL da proposta, respectivamente. Os relatórios publicados por esses grupos reafirmaram o relatório do DBTG em 1971, com algumas alterações e explicações. A técnica proposta de gerenciamento de banco de dados é, portanto, normalmente chamada de proposta DBTG ou sistema DBTG, e assim será referenciada no restante deste artigo.

Apesar da proposta DBTG empregar os elementos da linguagem que estão no estilo do COBOL e que foram obviamente especificados tendo-se COBOL em mente, eles poderiam ser acrescentados às outras linguagens com algumas modificações. Mas, possivelmente, os recursos de manipulação de dados estariam disponíveis nas outras linguagens como funções. A CODASYL, propriamente dita, não desenvolve ou fornece software mas, como o *MUMPS Development Committee* (que foi constituído nos moldes da CODASYL), ela encoraja fornecedores de software a seguir as suas propostas. Esta proposta pode se tornar de fato um padrão em virtude do respeito de que goza a CODASYL entre os órgãos governamentais e a indústria de computadores. Por essa razão, a proposta DBTG possivelmente se tornará a técnica de gerenciamento de dados mais comum, assim como o COBOL se tornou a linguagem de programação mais comum. A proposta do DBTG é também de grande interesse porque ela é um superconjunto de outros sistemas de gerenciamento de banco de dados.

Como no projeto do COBOL, um forte recurso para o processamento de dados foi imaginado e um enorme grupo de características foi incluído para satisfazer a uma grande comunidade de usuários. Poder e perfeição eram critérios muito mais importantes do que unidade ou elegância. Como resultado, é quase impossível encontrar falhas na proposta DBTG em relação à falta de algum recurso útil. Os pontos negativos possíveis estão nas áreas de complexidade (para o usuário e o implementador) eficiência e exigências maciças de hardware.

Não existe atualmente implementação que cumpra todos os requisitos da proposta DBTG, mas existem várias implementações comerciais que incluem um subconjunto substancial. Entre essas estão o DMS/1100 da Univac, o IDMS (para a série 360/370 da IBM) da Cullinane e DBMS-11, DBMS-10 e DBMS-20 para o PDP-11/70, PDP-10 e PDP-20, respectivamente, fornecidos pela Digital Equipment Corporation.

A proposta da DBTG lida com três tipos de entidades principais: registros, conjuntos e itens. A pessoa ou pessoas encarregadas do banco de dados (denominado administrador de banco de dados) definem cada tipo de registro até o nível

de campo. Elas definem as maneiras como os registros estão inter-relacionados através da definição de conjuntos. Estas definições compõem o "esquema". Também pode haver subesquemas que permitem que um conjunto específico de programas de aplicações causem uma versão modificada ou subconjunto do esquema inteiro.

Um conjunto tem um tipo de "registro dono" e um tipo de "registro membro". Um dado conjunto tem uma "ocorrência" para cada ocorrência do tipo de "registro dono". Assim, num sistema de controle de estoque, um conjunto chamado de "Fornecimento de item" pode ter a Companhia Fornecedora como dona e os registros de itens como membros. Cada Fornecedor e cada item teria um registro. Os registros de todos os itens fornecidos por um único fornecedor seriam membros de uma ocorrência do conjunto e aquele fornecedor seria o dono. Um registro pode ser o dono de um conjunto e um membro de outro, permitindo-se hierarquias. Também, um registro pode ser um membro de mais de um conjunto, permitindo-se redes. Infelizmente, um único registro não pode ser um membro de múltiplas ocorrências de qualquer conjunto (tais como um registro único de item pertencendo a múltiplos fornecedores). Esta restrição é contornada pela definição de registros de ligação.

A definição de um conjunto pode também oferecer opções tais como: (1) ordenação dos "registros membro" (isto é, se eles serão seqüenciados por ordem temporal na qual elas são acrescentadas ao conjunto, ou por alguma chave ou de alguma outra maneira); (2) se o sistema deve ou não rejeitar a tentativa de inclusão de duplicatas; (3) se os registros que satisfazem à condição de pertinência devem ou não ser automaticamente acrescentados ao conjunto quando são criados.

BANCO DE DADOS RELACIONAIS

Bancos de dados relacionais são uma forma de se organizar os dados em matrizes que teoricamente permitem a descrição de qualquer possível operação de pesquisa ou atualização numa maneira simples, mas rigorosa. A estrutura de banco de dados relacionais é atraente porque ela é adequada para linguagens não baseadas em procedimentos (non-procedural) nas quais o programador pode descrever o que ele quer que seja feito enquanto o sistema trata da maioria dos detalhes de como será realizado.

Tem existido um grande número de implementações experimentais de sistemas relacionais, mas não há indicação alguma quanto a esses sistemas serem suficientemente eficientes e práticos para um ambiente de produção. Apesar do trabalho mais recente nesta área parecer muito promissor, as aplicações comerciais têm sido limitadas a poucas implementações que não estão disponíveis para análise.

COMPARAÇÕES

Diferentes sistemas de gerenciamento de banco de dados podem ser comparados seguindo diversos critérios, incluindo suas capacidades, facilidades de uso, eficiência (em termos de velocidade ou armazenamento) e exigências de hardware. O primeiro desses é a comparação mais fácil de se fazer por causa da documentação publicada das capacidades de cada sistema. Os outros critérios tendem a ser mais subjetivos ou exigem informações que não estão disponíveis, mas nem por isso são menos importantes para um julgamento equilibrado.

O MUMPS tem excelente capacidade de gerenciamento de registros, incluindo registros de tamanho variável, chaves esparsas e indexação hierárquica. O MUMPS carece de duas características úteis que estão presentes em muitos outros sistemas de gerenciamento de registros: as chaves não numéricas (nota do tradutor: esta limitação não existe no MUMPS da MEDIDATA — vide nota anterior) e a capacidade de manipular facilmente itens de dados abaixo do nível do nodo/registro. Entretanto, esta capacidade é conseguida através do código de programas MUMPS.

O MUMPS compartilha com alguns sistemas de gerenciamento de banco de dados a capacidade de representar as estruturas de dados hierárquicas. Ele não manuseia redes, apesar de haver maneiras nas quais os programadores MUMPS possam simular redes. Ele na verdade não é um sistema de gerenciamento de banco de dados, no sentido de que não fornece dicionários de dados, definições de dados centralizadas e independência lógica de dados. De fato, as primeiras implementações do MUMPS foram sujeitas à dependência física de dados porque havia uma correspondência unívoca entre um grupo de nodos irmãos e uma cadeia de blocos de disco. Esta relação leva os projetistas de arquivos a condensarem níveis logicamente separados em um único nível, para evitar uma má utilização de disco, e a dividir um único nível lógico em níveis múltiplos, para conseguir recuperação rápida. Esta estrutura não é inerente no MUMPS e algumas implementações recentes rompem com este médoto. (nota do tradutor: é o caso do MUMPS da MEDIDATA).

O MUMPS é uma das poucas linguagens que foi projetada tendo-se em mente as estruturas complexas de dados. Assim sendo, ele manuseia esses campos de maneira bem menos estranha do que as linguagens que inseriram chamadas especiais para interfacear com os sistemas de gerenciamento de registros (métodos de acesso) ou gerenciamento de banco de dados. As extensões de manipulação de dados para o COBOL, proposta pelo DBTG e pelo DBLTG, são, entretanto, uma adaptação razoável para essa linguagem.

Todos os verdadeiros sistemas de gerenciamento de banco de dados exigem, de forma considerável, mais memória principal e/ou mais componentes de hardware do que o MUMPS. Até pouco tempo nenhum deles podia ser utilizado em um computador médio ou pequeno. Pelo menos dois (TOTAL e DBMS-11) podem ser utilizados agora em computadores de médio porte, mas exigem uma configuração mais cara do que uma instalação típica MUMPS. Por exemplo, o DBMS-11 exige um PDP 11/70 e pelo menos 256 Kbytes da memória principal, ao passo que o MUMPS pode ser executado eficientemente em sistemas de 64 Kbytes, ou menos.

Os sistemas de gerenciamento de registros, que fornecem limitada capacidade indexada hierárquica num computador médio ou pequeno, apareceram recentemente.

Os sistemas de gerenciamento de banco de dados tendem a ter muitos níveis de construção de sistema entre o programador de aplicações e a tarefa que ele quer que seu programa execute. Existe um esquema que define o banco de dados e um subesquema que redefine a parte do banco de dados usado pela aplicação especificada. O sistema de gerenciamento de banco de dados fica entre o programa de aplicações e o sistema operacional. Um programa freqüentemente passa por dois níveis de tradução: pré-processamento e compilação. O resultado é que as mensagens de erro durante o tempo de execução podem ser muito difíceis de entender. Também, como o sistema assume mais tarefas, o programador de aplicações tem menos capacidade para fazer ajustes finos visando a eficiência.

Uma característica vital que é inerente no método MUMPS é a sua facilidade de atualização do projeto básico de arquivos. Uma vez que a estrutura é definida pelo usuário numa linguagem interpretativa, é muito mais fácil modificar a estrutura da árvore de um arquivo usando MUMPS do que qualquer outro sistema DBMS, que exige uma especificação de atributos do arquivo como uma função separada, distinta do acesso ou manipulação dos elementos de dados.

O MUMPS, então, toma a posição intermediária entre um sistema de gerenciamento de banco de dados totalmente especificado e controlado e um método de acesso estritamente físico. Ele não inclui todas as características dos sistemas de gerenciamento de banco de dados maiores e mais completos. Entretanto, ele oferece um sistema básico sólido para a manipulação de grandes bancos de dados on-line, de uma maneira dinâmica e relativamente eficiente.

COMPARAÇÃO ENTRE MUMPS, COBOL, FORTRAN, PL/1, APL E BASIC

Exemplo de Problema

Para demonstrar como as diferentes linguagens se apresentam ao programador, um problema-exemplo foi codificado em cada linguagem. O problema-exemplo pode não ser uma demonstração exaustiva da força de cada linguagem, mas ele serve como referência rápida para os programas correspondentes.

O problema a ser codificado em cada linguagem é:

- Exiba a mensagem "ENTER PATIENT NUMBER" em um terminal.
- Recupere um número de paciente do terminal do usuário.
- Se a resposta tiver sido não-numérica, exiba a mensagem "NON-NUMERIC TRY AGAIN" e volte para a etapa 1.
- Se a resposta tiver sido numérica, solicite a uma sub-rotina que procure o número do paciente no arquivo principal e retorne o nome do paciente. Os detalhes internos dessa sub-rotina não são considerados neste programa.
- Exiba a mensagem "PATIENT NAME IS": seguida do nome do paciente.

Já que o COBOL, FORTRAN e PL/1, geralmente, exigem um monitor de comunicações (*) separado para o suporte de terminais interativos, assumiremos que tais capacidades estejam disponíveis, fornecendo interface aos programas com as seguintes especificações:

TPMSGIN — leia uma linha do terminal
TPMSGOUT — escreva uma linha no terminal.

(*) Nota do tradutor: tais monitores são CICS, ENVIRON, INTERCOM, etc.

Assim, temos:

VERSÃO MUMPS

```
DEMO; SAMPLE PROBLEM TO READ PATIENT NUMBER
AND
   DISPLAY PATIENT NAME
   READ! 'ENTER PATIENT NUMBER', PNUM
   IF PNUM! . ?N WRITE!, 'NON NUMERIC, TRY
   AGAIN' GOTO
   DEMO
   DO  ^ LKP; RETURNS PATIENT NAME IN
   PNAM WHITE!, 'PATIENT NAME IS', PNAM
   QUIT
```

VERSÃO COBOL

```
IDENTIFICATION DIVISION.
   PROGRAM-IDENTIFICATION. DEMO.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
   SOURCE-COMPUTER. IBM-370.
   OBJECT-COMPUTER. IBM-370.
DATA DIVISION.
WORKING-STORAGE SECTION.
   01 PATIENT-NUMBER    PICTURE 9(6).
   01 ERROR-MESSAGE     PICTURE X(20).
                        VALUE 'WRONG, TRY
                        AGAIN'.
   01 DISPLAY-LINE.
      05 FILLER PIC X(17) VALUE 'PATIENT NAME IS:'
      05 PATIENT-NAME PIC X(25).
   01 TP-PARMS PIC X(24).
   01 HDR-MSG PIC X(20) VALUE 'ENTER PATIENT
      NUMBER'.
PROCEDURE DIVISION.
GET-PATIENT-NUMBER.
   USING TP-PARMS HDR-MSG.
   CALL 'TPMSGIN' USING TP-PARMS
   PATIENT-NUMBER.
   IF PATIENT-NUMBER NOT NUMERIC
      CALL 'TPMSGOUT' USING TP-PARMS OF
      ERROR-MSG.
      GOTO GET-PATIENT-NUMBER.
   CALL 'LOOKUP' USING PATIENT-NUMBER
   PATIENT-NAME.
   CALL 'TPMSGOUT' USING TP-PARMS DISPLAY-LINE.
   GOBACK.
```

VERSÃO FORTRAN

```
      LOGICAL * 1 NAME(24), NUM(6), LO/'0'/,HI/'9'/
      INTEGER BEG(5)/'ENTE', 'R PA', 'TIEN',
     "T NU', "MBER'/
      INTEGER ERR(6)/'NON-', 'NUME', 'RIC,', 'TRY', 'AGAI',
     'N'/
      INTEGER END(10)/'PATI', 'ENT', 'NAME', 'IS:'/
      EQUIVALENCE (NAME, END(5))
      INTEGER TPPARM(5)

C     THIS PROGRAM WILL READ PATIENT NUMBER AND
      DISPLAY
      PATIENT NAME
    1 CALL TPMSGOUT(TPPARM, BEG)
      CALL TPMSGIN(TPPARM, NUM)
      DO 2 I = 1, 6
      IF NUM(I) . LT . LO . OR . NUM(I) . GT . HI) GO TO 400
    2 CONTINUE
      CALL LOOKUP (NUM, NAME)
      CALL TPMSGOUT(TPPARM, END)
      RETURN
  400 CALL TPMSGOUT(TPPARM, ERR)
      GO TO 1
      END
```

VERSÃO PL/1

```
DEMO: PROC OPTIONS (MAIN);
/ * SAMPLE PROBLEM TO READ PATIENT
NUMBER AND
DISPLAY NAME * /
DECLARE
PATNUM CHARACTER(6),
PATNAME CHARACTER(25),
TPPARMS CHARACTER(20); / * MISC MONITOR
PARAMETERS*/
LOOP:    CALL TRMSGOUT(TPPARMS, 'ENTER
         PATIENT NUMBER');
         CALL TPMSGIN(TPPARMS, PATNUM);
         IF NOT VERIFY (PATNUM,"0123456789')
         THEN DO;
         CALL TPMSGOUT(TPPARMS, 'NON-NUMERIC,
         TRY AGAIN');
         GOTO LOOP;
         END;
         CALL LOOKUP(PATNUM, PATNAME);
         CALL TPMSGOUT(TPPARMS, 'PATIENT NAME
         IS: 'CAT
         PATNAME);
         RETURN;
         END DEMO;
```

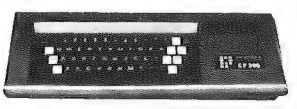
SOLUÇÃO NÃO É PROBLEMA

não importa o tamanho de seu problema, nós temos a solução na medida exata!

CP-200 COM SPEED

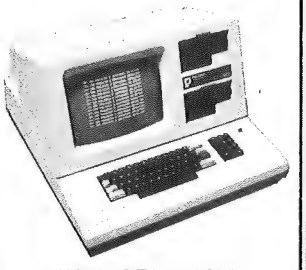
- LINGUAGEM BASIC
- 16 K DE MEMÓRIA
- VELOCIDADE DE TRANSFERÊNCIA 14 VEZES MAIS RÁPIDA

CP-300



- MODULAR
- LINGUAGEM BASIC
- 48 K DE MEMÓRIA
- COMPATÍVEL COM SOFTWARE DO CP-500

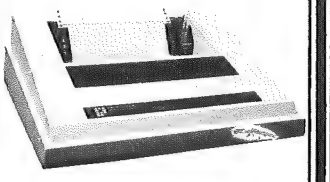
CP-500



- LINGUAGEM BASIC
- 48 K DE MEMÓRIA
- ATÉ 4 DRIVES
- SAÍDA PARALELA SERIAL

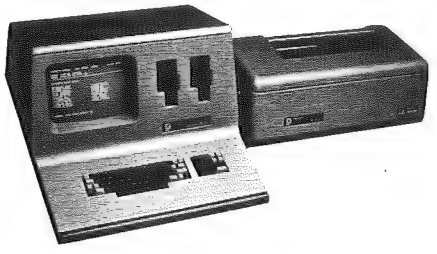
P-500

JÁ A VENDA



- VELOCIDADE 100 CPS
- MATRIZ 9 x 7
- INTERFACE: PARALELA SERIAL

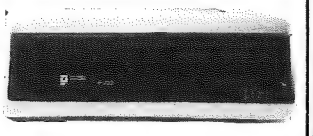
S-600



MICRO:
- LINGUAGENS COBOL, BASIC E FORTRAN
- 64 K DE MEMÓRIA
- DUAS UNIDADES DE DISCO

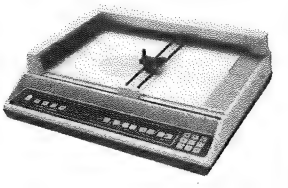
IMPRESSORA:
- VELOCIDADE 130 CPS
- MATRIZ 7 x 9
- 132 COLUNAS
- ORIGINAL +5 CÓPIAS

P-720



- VELOCIDADE 200 CPS
- MATRIZ 7 x 9
- INTERFACE: PARALELA, SERIAL

TRAÇADOR GRÁFICO



- 8 PENAS
- ÁREA DE TRAÇADO 10 x 15 POL.
- INTERFACE RS-232

ACESSÓRIOS

SOFTWARE • MESAS • DISQUETES • ARQUIVOS • FORMULÁRIOS CONTÍNUOS • ESTABILIZADORES DE TENSÃO • UNIDADES DE DISCO FLEXÍVEL • ETC.

APROVEITE!

PROMOÇÕES ESPECIAIS • FINANCIAMENTO • LEASING • CONSÓRCIO • CARTÕES DE CRÉDITO: CREDICARD, NACIONAL, ELLO.

filcres

Filcres Importação e Representações Ltda.
Rua Aurora, 165 – CEP 01209 – São Paulo – SP
Telex 1131298 FILG BR – PBX 223-7388 – Ramais 2, 4, 12, 18, 19 – Diretos: 223-1446, 222-3458, 220-5794 e 220-9113 - Reembolso – Ramal 17 Direto: 222-0016 – 220-7718

VERSÃO APL

```
[1]    "ENTER PATIENT NUMBER"
[2]    PNUM
[3]→ (PNUM < '0') v (PNUM > '9')/ERR
[4]    'PATIENT NUMBER IS: ', LOOKUP
[5]→ 0
[6]    ERR: "NON-NUMERIC, TRY AGAIN"
[7]→ 1
```

VERSÃO BASIC

```
 05   DIM P$ (6), N$ ("£)
 10   PRINT "ENTER PATIENT NUMBER"
 20   INPUT P$
 30   REMARK TEST EACH DIGIT TO SEE NUMERIC
 40   FOR I = 1 TO 6
 50   IF P$ (I, I) < "0" GOTO 400
 60   IF P$ (I, I) > "9" GOTO 400
 70   NEXT I
 80   REMARK LINE 1000 IS THE SUBROUTINE TO
      RETURN THE PATIENT NAME IN N$
 90   GOSUB 1000
100   PRINT "PATIENT NAME IS:", N$
110   STOP
400   PRINT "NON-NUMERIC, TRY AGAIN"
410   GOTO 10
```

QUADROS COMPARATIVOS

Cada um dos comandos, funções e operadores MUMPS são listados junto com os correspondentes mais parecidos nas outras linguagens. Os diagramas têm a intenção de apresentar o MUMPS a um programador familiarizado com uma das outras linguagens. Portanto, os elementos das outras linguagens que não correspondem a qualquer um dos elementos em MUMPS foram omitidos.

As funções $NEXT e $DATA do MUMPS são comparadas com IMS, o Sistema de Gerenciamento de Informações da IBM, a fim de comparar as duas com suas capacidades de manipulação de arquivos. Deveríamos notar que o resto das funções IMS manipuladas pelo processador DL/1 (Data Language 1) são manipuladas em MUMPS pelos comandos de armazenamento de globais SET e KILL.

Nota do tradutor: o conteúdo dos quadros a seguir não foi traduzido, já que envolve termos que são amplamente utilizados pela comunidade de PD no Brasil.

OS COMANDOS MUMPS EM OUTRAS LINGUAGENS

MUMPS	COBOL	FORTRAN	PL/1	APL	BASIC
SET X = Y	MOVE Y to X	X = Y	X = Y	X ← Y	LET X = Y
KILL X	—	—	—	)ERASE X	—
IF X = Y	IF X IS EQUAL Y	IF (X . EQ . Y)	IF X = Y	—	IF X = Y
FOR I = X: Y: Z	PERFORM *paragraph* VARYING I FROM X TO Z BY Y	DO number I = X, Z, Y	DO I = X TO Z BY	—	FOR I = X TO Z STEP Y
GOTO label	GOTO *paragraph*	GOTO *number*	GOTO *label*	→ *label*	GOTO *number*
GOTO ^ *label*	—	—	—	—	CHAIN *program*
DO *label*	PERFORM *paragraph*	—	—	—	GOSUB *number*
DO ^ *label*	CALL *program*	CALL *program*	CALL *program*	function reference	—
QUIT	GOBACK, EXIT	RETURN	RETURN	→ 0	RETURN
HANG	—	—	DELAY	—	—
HALT	STOP RUN	STOP	STOP	)OFF	BYE
BREAK	—	—	—	S Δ P	—
READ X	READ . . . INTO X	READ (. .) ,X	GET . . . (X)	X ← □	INPUT X
WRITE X	WRITE . . . FROM X	WRITE (. .) ,X	PUT . . . (X)	□ ← X	PRINT X
XECUTE	COPY*	—	%INCLUDE*	—	—
LOCK	—	—	EXCLUSIVE, UNLOCK	—	—
OPEN	—	—	—	—	—
CLOSE	—	—	—	—	—
USE	—	—	OPEN . . . TITLE	—	—

AS VARIÁVEIS ESPECIAIS E FUNÇÕES EM OUTRAS LINGUAGENS

MUMPS	COBOL	FORTRAN	PL/1	APL	BASIC
$DATA	(fully qualified Get	Unique call)**		—	—
$NEXT	(Get Next call)**			—	—
$LENGTH	—	—	LENGTH	rho	LEN
$FIND	—	—	INDEX	iota	—
$EXTRACT	—	—	SUBSTR	subscripting	subscripting
$PIECE	—	—	—	—	—
$TEXT	—	—	—	—	—
$JUSTIFY	PICTURE	FORMAT	EDIT	—	PRINT USING
$ASCII	—	—	—	—	—
$CHAR	—	—	—	—	—
$SELECT	—	—	—	—	—
$RANDOM	—	—	—	roll	RND
$IO	—	—	—	I-beam 28	—
$X, $Y	—	—	—	—	—
$HORLOG	TIME-OF-DAY CURRENT-DATE	—	TIME, DATE	I-beam 20, 25	SYS (0, 1, 2, 3)
$SIZE	—	—	—	I-beam 22	—
$TEST	—	—	—	—	—

** IMS (IBM's INFORMATION MANAGEMENT SYSTEM) SUPPORT OF THESE LANGUAGES

OPERADORES MUMPS EM OUTRAS LINGUAGENS

MUMPS	COBOL	FORTRAN	PL/1	APL	BASIC
+	ADD	+	+	+	+
–	SUBTRACT	–	–	–	–
*	MULTIPLY	*	*		*
/	DIVIDE	/	/		/
\	DIVIDE . . ROUNDED	integer mode	integer mode	–	–
#	–	–	MOD	residue	–
<	LESS THAN	<	<	<	<
>	GREATER THAN	>	>	>	>
(NOT)	NOT	.NOT.		tilde	–
– (CONCATENATION)	–	–	CAT		'
] (FOLLOWS)	>	>	>	–	>
[(CONTAINS)	–	–	INDEX	–	–
? (PATTERN MATCH)	–	–	–	–	–
& (AND)	–	.AND.	AND	∧	–
! (OR)	OR	.OR.	OR	∨	–
@ (INDIRECTION)	–	–	–	–	–
: (POST CONDITIONAL)	–	–	–	–	–

CONCLUSÃO

Este relatório fez uma tentativa de enfocar as características da linguagem MUMPS, sistema operacional e sistema de banco de dados. Ele também comparou esses aspectos com alguns outros sistemas hoje existentes. A comparação não conseguiu ser completa, já que o campo de processamento de informações está em constante expansão e uma análise abrangente está muito além do escopo desse relatório. O autor espera que as discussões anteriores tenham afirmado corretamente os pontos fortes e negativos do MUMPS em comparação com outros sistemas.

O MUMPS não é um sistema para todos os usuários ou todas as aplicações. Ele é um instrumento muito útil e eficiente para o desenvolvimento e suporte de sistemas interativos de gerenciamento de dados. Uma característica universal de tais sistemas é que seus objetivos e procedimentos variam com o tempo. Em parte, essas mudanças podem ser relacionadas com inadequações nos estágios iniciais de planejamento. Em muitos casos, entretanto, eles estão relacionados com o efeito psicológico da introdução da tecnologia do computador em um novo campo. É um prêmio para uma aplicação bem-sucedida, o fato dos usuários modificarem sua atitude inicial, elevarem suas expectativas e fazerem pedidos cada vez maiores de serviços de manipulação de informações. O sistema MUMPS enfrenta esse desafio com um *approach* integrado para o desenvolvimento de sistemas de aplicações. A combinação de uma linguagem interpretada e de alto nível e um sistema de gerenciamento de banco de dados dinâmico e hierárquico provou ser um meio eficiente e eficaz para a implementação de sistemas de informação. Se, de um lado, se reconhece que o MUMPS não pode resolver todos os problemas de gerenciamento de informações, sabe-se, por outro lado, que ele oferece características únicas e significativas que deveriam ser consideradas em qualquer avaliação que se fizesse de sistemas de informação.

BIBLIOGRAFIA

- GREENES, R.A., PAPPALARDO, A. N., MARBLE, C.W., and BARNETT, G.O (1969) Design and Implementation of a Clinical Data Management System, Computers in Biomedical Research 2, 469-485.
- BOWIE, J., and BARNETT, G.O. (1976) MUMPS-An Economical and Efficient Time-Sharing System for Information Management, Computer Programs in Biomedicine. 6.11-22.
- Proceedings of 1973 MUMPS Users' Group Meeting, St. Louis, Missouri
- Proceedings of 1974 MUMPS Users' Group Meeting. Denver, Colorado.
- O'NEILL, J.T. (ed.) (1975) MUMPS Language Standard, National Bureau of Standards Handbook 118 (Washington, D.C.).
- JOHNSON, M.E. and DAYHOFF, R.E. 1975, MUMPS Primer, MDC 1/11, MUMPS Development Committee, 1975.
- EGERMAN, P. L., LAZARUS, C.B., and RAGON, P.T. (1975) MUMPS Translation Methodology, MDC/30, MUMPS Development Committee. 1975.
- IVERSON, K.E. (1962) A Programming Language (Wiley, New York).
- ISACCS, G. L. (1974) Interdialect Translatability of the BASIC Programming Language, ACM Sigcue Bulletin 8, (4), 11-12.
- DATE, C. J. (1975) An Introduction to Database Systems (Reading, MA: Addison Wesley Publishing CO.).
- CODASYL Systems Commitee, A Survey of Generalized Data Base Management Systems (Available from ACM) May, 1969.
- Codd E.F. (1970) A Relational Model of Data for Large Shared Data Banks, CACM 13, Nº 6.
- Codd E.F. (April 1974) Seven Steps to Rendezvous with Casual User, Proc. IFIP TC-2 Working Conference on Data Base Management Systems.
- ACM Computing Surveys: Special Issue on Data Base Management, 8,1, March, 1976.
- BAREFORD, J.L. (16 Nov 1976) Personal Communication to R.F. Walters
- DIJKSTRA, E. W (1972) The Humble Programmer, Communications of the ACM, 15, 859-866.

ADVANCING
CONSULTORIA DE PESSOAL E TREINAMENTO

ADVANCING
COMPUTER SHOP

ADVANCING
SOFTWARE HOUSE

EMPRESAS DO GRUPO ADVANCING:
CONSULTORIA - TREINAMENTO - DIREÇÃO Andradas 1560 - Cj. 518 - 5º and.
COMPUTER SHOP - SOFTWARE HOUSE Sarmento Leite, 248 - Fones (0512) 26-8246/26-0194/26-1194 P. Alegre - RS

A ATUAL POLÍTICA DE INFORMÁTICA BRASILEIRA

Reinaldo de Freitas Pires de Moraes
Contador
Rua Parnapoã, 374
São Paulo – SP

A ATUAL POLÍTICA DE INFORMÁTICA BRASILEIRA

Este pequeno trabalho é, em síntese, um grande ponto de interrogação, pois o próprio tema o é.

Em todos os simpósios e debates já realizados em torno do assunto nada se concluiu e sempre uma série de perguntas ficam com respostas pendentes. Porém, dois assuntos se sobressaem.

O primeiro é em relação a abrangência da Secretaria Especial de Informática e o segundo referente a eficácia da reserva de capital para as indústrias de capital nacional.

Para a primeira questão não temos resposta totalmente convincente, ou seja, quais os reais objetivos desta secretaria? Será a implantação de uma política NACIONAL ou de uma política GOVERNAMENTAL de informática?

Não podemos nos esquecer que atrás da SEI existe outra sigla parecida, porém, vista com certo receio: o S.N.I.

Esperamos que no futuro não ocorra a fusão dos dois organismos, aproveitando-se letras comuns às duas siglas, e dos recursos da informática, objetivando a confidencialidade de dados, a decretação de normas inflexíveis e a centralização decisória sem consulta aos envolvidos e normalmente partindo de gabinetes fechados.

Quanto à reserva de mercado: – estará ela voltada para a defesa somente dos fabricantes de minis e micros equipamentos? De que adianta uma política protecionista deste único segmento se ficarmos na dependência dos fabricantes de componentes eletrônicos e esses dos fornecedores de matérias-primas?

O campo da informática é abrangente e não se restringe somente ao equipamento final acabado. Além da matéria-prima e dos subprodutos, temos ainda os fabricantes de material de uso, de periféricos, assistência técnica e evidentemente a prestação de serviços e a própria indústria de Software.

A reserva de mercado é necessária, porém, ao adotá-la não devemos esquecer que as empresas envolvidas terão que desenvolver tecnologia própria, ter o dinamismo para absorver e desenvolver a tecnologia criada diariamente em todo o mundo e lembrar que o capital aplicado tem seu retorno a médio e a longo prazo, apesar de que por ocasião deste retorno e rentabilidade é sempre vantajosa.

Não podemos admitir multinacionais atuando através de testas-de-ferro absorvendo toda empresa lucrativa ou em ramos que por lei são restritos a capitais nacionais.

Devemos reivindicar a garantia de compra por parte do governo, de uma parcela do equipamento fabricado, taxação do equipamento importado, subsídios às indústrias e benefícios fiscais.

Precisamos capacitar o país a desenvolver e produzir equipamentos, nacionalizar os insumos básicos, estabelecer normas técnicas, dando também a mesma atenção ao Software, pois não pode haver indústria nacional de Soft em cima de equipamentos estrangeiros.

Porém, o que considero mais importante é a aplicação de recursos na pesquisa tanto de Soft como de Hardware.

Podemos chamar de "Indústria Nacional" se toda tecnologia aplicada for estrangeira? Podemos qualificar um produto de "Made in Brazil" se este produto é uma cópia fiel dos importados?

Nossas universidades e centros de pesquisas já provaram que são capazes, não devendo nada a ninguém. Estaremos sempre no mesmo nível de igualdade, se suficientes verbas forem destinadas a nossos pesquisadores, até então em número pequeno. Possuímos pessoal com qualidade internacional, mas que infelizmente deixam o país em busca de melhores salários e novos conhecimentos, ou são deslocados para vendas.

Outro fator a ser lembrado são os custos de fabricação. Mundialmente os preços são reduzidos. No Brasil, excluindo o "negativismo da crise", poderemos chegar a custos e preços compatíveis.

Com a reserva de mercado muitas empresas surgiriam, levando fatalmente à competição interna que resultaria na diminuição dos custos de fabricação e serviços com respectiva melhora de qualidade.

Porém, nem todos são a favor da reserva de capital.

Os contrários propõem a constituição de "joint-ven-

tures" para a exploração desta área. Consideram nossa indústria muito jovem e carente de retaguarda sólida. Afirmam que o Estado não deve, com excessos de regulamentações, inibir o desenvolvimento das indústrias e que a proteção perene traz consigo o risco da obsolescência. Não acredito que a indústria nacional possa sofrer um isolamento tecnológico.

Os que são contra a reserva são pessoas normalmente ligadas a empresas ou grupos estrangeiros.

As multinacionais estão cientes que o mercado de informática no Brasil é um dos mais promissores do mundo, daí sua pressa em abocanhar o mais rápido possível o melhor pedaço do bolo.

A indústria nacional deve se conscientizar que tem responsabilidades pelo seu próprio desenvolvimento tecnológico, senão estará vulnerável a pressões externas como as que hoje são feitas, contrariando a opinião de que este desenvolvimento feito com nossas próprias capacidades será moroso.

É bom lembrar que um documento do Departamento de Comércio dos EUA enfatiza a necessidade de mudanças na nossa política de informática, para assegurar a cooperação entre os dois países, pois o desejo do governo norte-americano é ver o Brasil mais receptivo à entrada de capitais de risco nas áreas que demandam alta tecnologia. Não devemos nos esquecer que a indústria nacional sofre uma concorrência desleal com as multinacionais, portanto, é necessário definir as regras do jogo.

Na área de serviços o próprio governo se torna um concorrente desonesto. É o caso da Datamec, pertencente a Caixa Econômica Federal, que vem ocupando espaços fora do campo para que foi criada, concorrendo diretamente com empresas nacionais que operam no setor.

Em síntese, todos falam, discutem e divergem. O governo, a indústria nacional e a indústria estrangeira, os pesquisadores, etc . . . sem ouvir, no entanto, o principal personagem: — o usuário, o qual vai pagar e usufruir dos serviços e equipamentos.

Se vivemos num clima de abertura política todos devem opinar, sempre lembrando que Democracia também tem como sinônimo a palavra Informação.

Este trabalho é o resultado do concurso do TK-82C, e pertence ao leitor Reinaldo de Freitas Pires de Moraes.

ERRATA — DICAS DE HARDWARE

Revista nº 11, página 31
FIGURA 1 — LED METER

os diodos D1 e D2 são invertidos, um em relação ao outro.

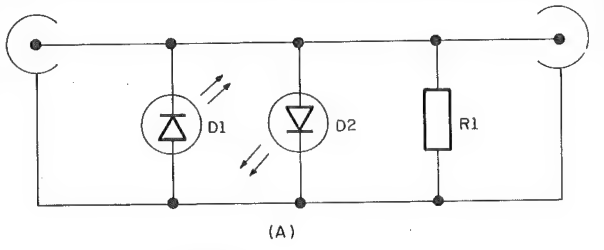


FIGURA 1 — LED METER

DISPONÍVEL PARA O SEU TK, ZX81, CP-200 E NE.
SOFTWARE E HARDWARE EXCLUSIVOS NO BRASIL

SOFTWARE

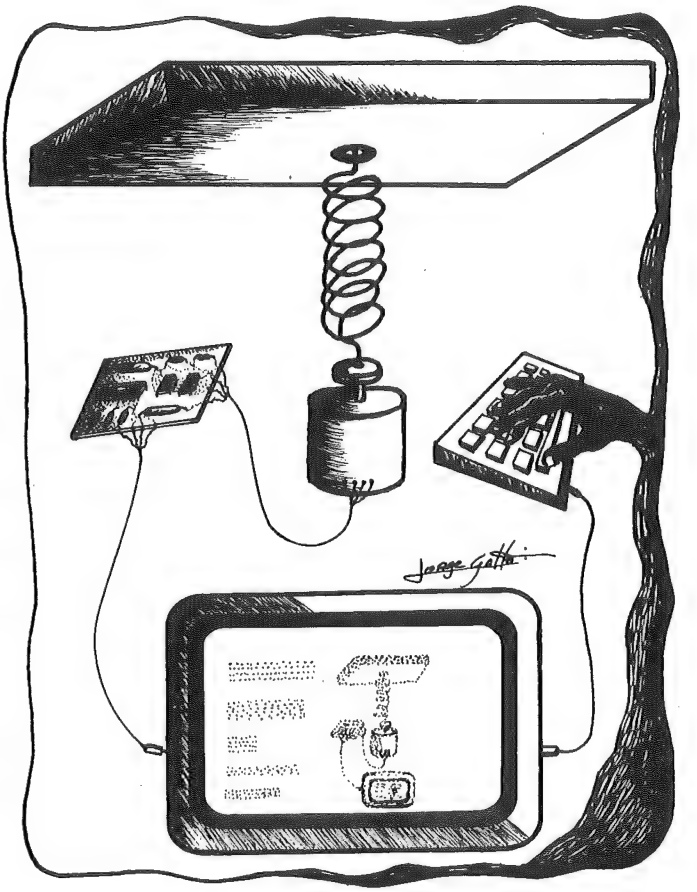
- SCREEN - 27 rotinas para gerenciar a tela, com funções e efeitos sensacionais. Cr$ 14.900,00.
- DOUBLE - Dobra a velocidade do SLOW, sem "Piscar a tela". Cr$ 5.500,00.
- MUSIC - Utiliza a saída MIC do seu micro, para a saída de som. Necessita amplificador externo. Cr$ 4.950,00.
- QUICK - Carrega, salva e verifica programas em fita, com o dobro da velocidade normal. Cr$ 14.500,00.
- COMPILADOR BASIC - Roda seus programas em BASIC, 60 vezes mais rápido. Cr$ 26.100,00.
- EDITOR/DISASSEMBLER — A linguagem de máquina sob seu comando. Cr$ 24.500.00.

HARDWARE

- K-7 CONTROLLER - Controla o motor do gravador automaticamente. Cr$ 16.500,00.
- RESET - Chave "reset". Cr$ 4.950,00.
- INVERT - Inversão (com chave) da imagem. Cr$ 5.200,00.
- INTER 200 - Permite que o CP-200 utilize a TK PRINTER ou outros periféricos. Cr$ 16.800,00.
- JOY-FACE - Conexão para JOY-STICK. Cr$ 14.300,00.
- EXTENSOR - Afasta a expansão de memória, evitando os desligamentos constantes. Cr$ 24.750,00
- MONITOR - Conexão direta do seu micro a qualquer monitor de vídeo. Cr$ 24.900,00.

ELDORADO COMPUTADORES E SISTEMAS LTDA.
Rua Visconde de Pirajá, 351 Ljs. 213/214
22410 - IPANEMA - RJ - Tel.: (021) 227-0791

O SOFTWARE NÓS REMETEMOS PARA TODO O BRASIL.



O SISTEMA OPERACIONAL UNIX

Por Ivan Costa e Mônica Fróes Peixoto

A LINGUAGEM "C"

O sistema operacional UNIX está intimamente ligado a linguagem *"C"* e este vínculo e uma das causas de sua força. *"C"* é uma linguagem projetada para a construção de programas rápidos, compactos e portáveis, podendo ser considerada como uma linguagem intermediária entre linguagens de alto nível tais como Pascal, Algol, Fortran e Basic e linguagens tipo Assembler, específicas para um determinado processador.

Simplicidade, elegância e potência são as características principais desta linguagem que oferece uma alternativa para padronização em um mercado caracterizado por grande diversidade de fornecedores e conseqüentemente de arquiteturas de processadores.

Com o advento das máquinas de 16 bits com códigos incompatíveis com seus populares antecessores de 8 bits a indústria de software sentiu a necessidade de padronização independente do hardware utilizado e *"C"*, por suas características que a tornam uma ótima ferramenta de desenvolvimento de sistemas, tem sido a escolha natural de *software houses*. Um exemplo deste fato é que a Digital Research Inc., que através do CP/M domina o mercado de sistemas de 8 bits e também retém significativa parcela do mercado de 16 bits, anunciou que visando fornecer portabilidade a nível de código fonte todas as novas versões do CP/M serão escritas em *"C"*.

Um *assembler portável* como chamada por Thomas Plum (1), *"C"* combina vantagens de várias linguagens e o número de compiladores disponíveis hoje para uma grande variedade de máquinas comprova o seu sucesso.

"C" foi desenvolvida nos laboratórios Bell por Denis Ritchie em 1972. Teve como base a linguagem *"B"*, escrita por Ken Thompson visando dar portabilidade ao UNIX. Tanto a linguagem *"B"* como *"C"* derivam da linguagem BCPL e mantêm características em comum com a mesma.

A linguagem *"C"* tem sido desde o seu começo ligada ao sistema UNIX. Escrito originariamente em assembler o UNIX foi reescrito por Ken Thompson e Denis Ritchie em *"C"*, tão logo esta ficou disponível.

Somente algumas poucas rotinas foram deixadas em assembler, e a partir daí a tarefa de transportar o UNIX para uma máquina em particular tornou-se basicamente a tarefa de se escrever um compilador *"C"* para a mesma. A maioria dos programas que compõe o UNIX executa sem problemas em um novo ambiente de hardware e estas características tornam o UNIX e *"C"* tão populares hoje.

POR QUÊ "C"?

A fase mais importante na implementação de um ambiente de programação é a escolha da linguagem de programação a ser utilizada. Este fato decorre de que o custo para uma eventual conversão de programas para uma outra linguagem é muito alto. A primeira opção será entre linguagens de alto nível ou linguagens assembler. Programas em linguagem assembler são teoricamente mais rápidos do que programas escritos em linguagem de alto nível, que têm de passar por um processo de conversão, resultando em um código quase sempre menos eficiente do que aquele gerado por um programador assembler experiente. Em contrapartida, a utilização de linguagem assembler acarreta uma série de desvantagens tais como:

- Tamanho do código fonte (programas escritos em assembler têm tamanho cerca de dez vezes maior do que o equivalente escrito em uma linguagem de alto nível).
- Complexidade na programação.
- São específicas para determinado processador (não são portáveis).

Estas características acarretam uma queda na produtividade em um ambiente de programação.

A opção por uma linguagem de alto nível e então quase que obrigatória, mas entre as disponíveis no mercado, qual a melhor opção?

As linguagens mais populares em micros são o Basic e Pascal. As mais utilizadas em computadores de grande porte: Cobol, Fortran e PL/1 são complexas e só recentemente implementações completas destas linguagens se tornaram disponíveis em micros. Existem outras linguagens que podem ser encaradas como opção: C, LISP, APL e FORTH.

LISP e APL são linguagens poderosas mas infelizmente não existem implementações completas de nenhuma das duas em micros. FORTH é uma linguagem nova, mas com uma grande quantidade de dialetos, sem um padrão seguro definido. As opções mais interessantes são, então, *"C"*, Basic e Pascal. As razões que levam a optar por *"C"* são as seguintes:

- Ela é significativamente mais rápida que o Basic e Pascal.
- Das três linguagens ela é a que apresenta maior grau de padronização e portabilidade.
- *"C"* possui suporte para programação estruturada (recursos encontrados em Pascal mas não em Basic).

Para avaliar a preferência por *"C"*, mesmo em ambientes não UNIX, podemos citar a existência de 5 compiladores *"C"* para o CP/M 80, 6 compiladores *"C"* para o CP/M 86 e 9 compiladores para o PC-IBM.

O MODELO "C"

A linguagem *"C"* se caracteriza por oferecer um modelo básico bastante próximo da arquitetura da máquina (permitindo escrever código bastante eficiente quando necessário) e ao mesmo tempo possibilitar a construção de modelos próprios através de mecanismos poderosos embutidos na linguagem.

Para aproximação com a arquitetura da máquina, *"C"* possui um grande número de operadores relacionados com as operações básicas encontradas em um processador. Deste modo, além das operações aritméticas usuais — soma (+), subtração (—), multiplicação (*) e divisão (/), existem operadores para incremento (++), decremento (— —), deslocamento à esquerda (<<) e à direita (>>) e também operadores lógicos — AND, OR, EXCLUSIVE OR e COMPLEMENTO A 1.

A dificuldade inicial para o aprendizado e manipulação destes operadores é recompensada pela versatilidade oferecida ao programador pelos mesmos. Além destes, existem em *"C"* operadores específicos para manipulação de bits. Estas operações de fundamental importância em programação de sistemas (p. ex. em operações de controle de dispositivos de entrada e saída) são facilmente programadas em *"C"* devido a existência destes operadores. Este recurso não é disponível em diversas linguagens, inclusive em Pascal.

"C" apresenta uma grande variedade de tipos de dados incluindo entre eles: bytes, inteiros curtos (tipicamente 16 bits), inteiros longos (tipicamente 32 bits), números de ponto flutuante em precisão simples e dupla.

O CONCEITO DE POINTER

Em *"C"* temos, como no Pascal e Algol, a existência de *pointers* para outro dado. Estes *pointers* correspondem a endereços de máquina e, como a maioria das instruções, manipula diretamente estes endereços. A existência de *pointers* encoraja a geração de código extremamente eficiente em segmentos críticos de programas bem como a construção de estruturas de dados altamente eficientes.

O modelo de *pointer* encontrado em *"C" inclui as* operações básicas de armazenamento de um endereço de uma variável (&variável) e acesso a uma variável apontada por um *pointer* (* pointer).

Por exemplo, suponha que *"X"* é um inteiro e *"P"* um *pointer* para um inteiro (não existe um *pointer* isolado; ele se apresenta sempre associado a um tipo de dado), então poderíamos ter a seqüência (2):

```
X = 10; /* atribua a X o valor 10 */
p = &X; /* atribua ao pointer p o endereço de X */
```

O conteúdo de *"p"* é o endereço de *"X"* e o valor apontado em *"p"* é referido por *" * p"*. Neste caso *" *p = 10"*. *" *p"* pode ser usado para alterar o valor de X:

```
*p = 4; /*X passa a ter o valor 4 */
```

É também possível utilizar o valor associado a um *pointer* para atribuição a outro. Se *"q"* também é um *pointer* para um inteiro, então:

```
q = p; /* q passa a apontar para onde p aponta */
       /* ver representação gráfica na fig. 1 */
       /* ver referência 2 */
```

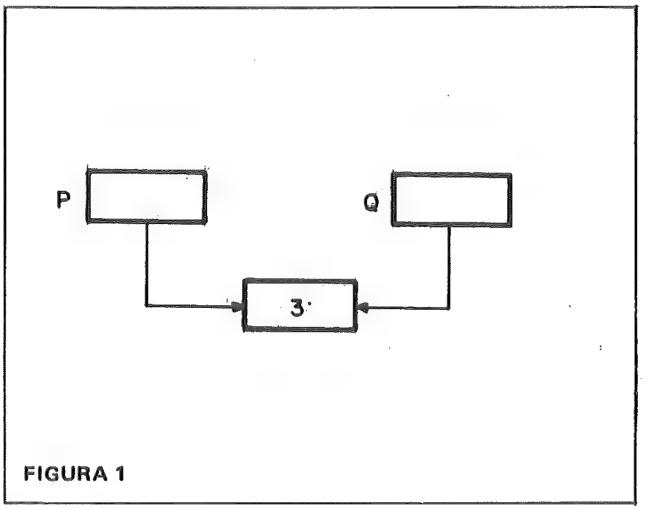


FIGURA 1

Além destas possibilidades, que são basicamente as encontradas em Pascal, *"C"* inclui a capacidade de manipulação de *pointers* em função do tipo de dado apontado. Por exemplo, se temos um *pointer* para um byte e incrementamos este, ele passa a apontar para o próximo byte. Se temos um *pointer* para um inteiro de 16 bits e incrementamos este, ele passa a apontar para o próximo inteiro (2 bytes adiante). *Pointers* são uma excelente ferramenta para indexação de *"arrays"* como mostrado na figura 2, onde X é um *array* de inteiros. (2)

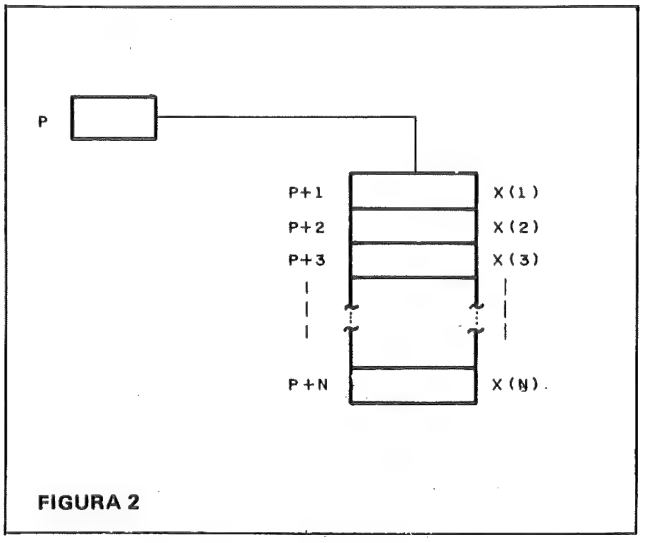


FIGURA 2

ESTRUTURA DA LINGUAGEM

Os programas em *"C"* são constituídos de funções e obrigatoriamente possuem a função *main*. Comandos compostos são delimitados pelos caracteres abre chave ({) e fecha chave (}) em substituição aos tradicionais *BEGIN* e *END* do Pascal e Algol. Esta é uma das características que torna os programas em *"C"* difíceis de serem lidos (e mantidos) pois como *"C"* tem formato livre (as linhas não ne-

cessitam de numeração), a colocação indevida ou esquecimento de um destes caracteres introduzem facilmente erros no programa. Outro fator de dificuldade para o programador iniciante é a possibilidade de um teste lógico ter como argumento uma expressão a ser valorizada na execução, por exemplo, a seqüência:

```
IF ( ( VAR = GETCHAR ( )) = = 'X') COUNT + + ;
```

A função *GETCHAR* lê um caractere e este é atribuído à variável *"VAR"*. Após isto é realizado o teste de igualdade entre a variável *"VAR"* e o caractere 'X', tudo isto dentro do teste condicional do comando IF. (3)

A tendência em programação *"C"* é reduzir programas para uma coleção de funções discretas que reduzem o esforço de programação e produzem código de fácil manutenção, ou seja, programas grandes são "quebrados" em programas menores e cada um destes é associado a uma função. Se cada função é independente, o programador pode desenvolver uma "Biblioteca" de funções que podem ser combinadas de diferentes maneiras para formar novos programas.

Um programa em C poderia ser:

```
/* EXP1.C exemplo de um programa em C */
main ( ) /* comentário */
{
}
```

Main é a função principal e obrigatória. Comentários são delimitados por "/* " e " */", e são opcionais.

Dentro de uma função podem existir chamadas para outra, como mostrado abaixo:

```
/* EXP2.C  chamada de uma função */
main ( )
{
fun ( ); -
}
fun ( )
{
}
```

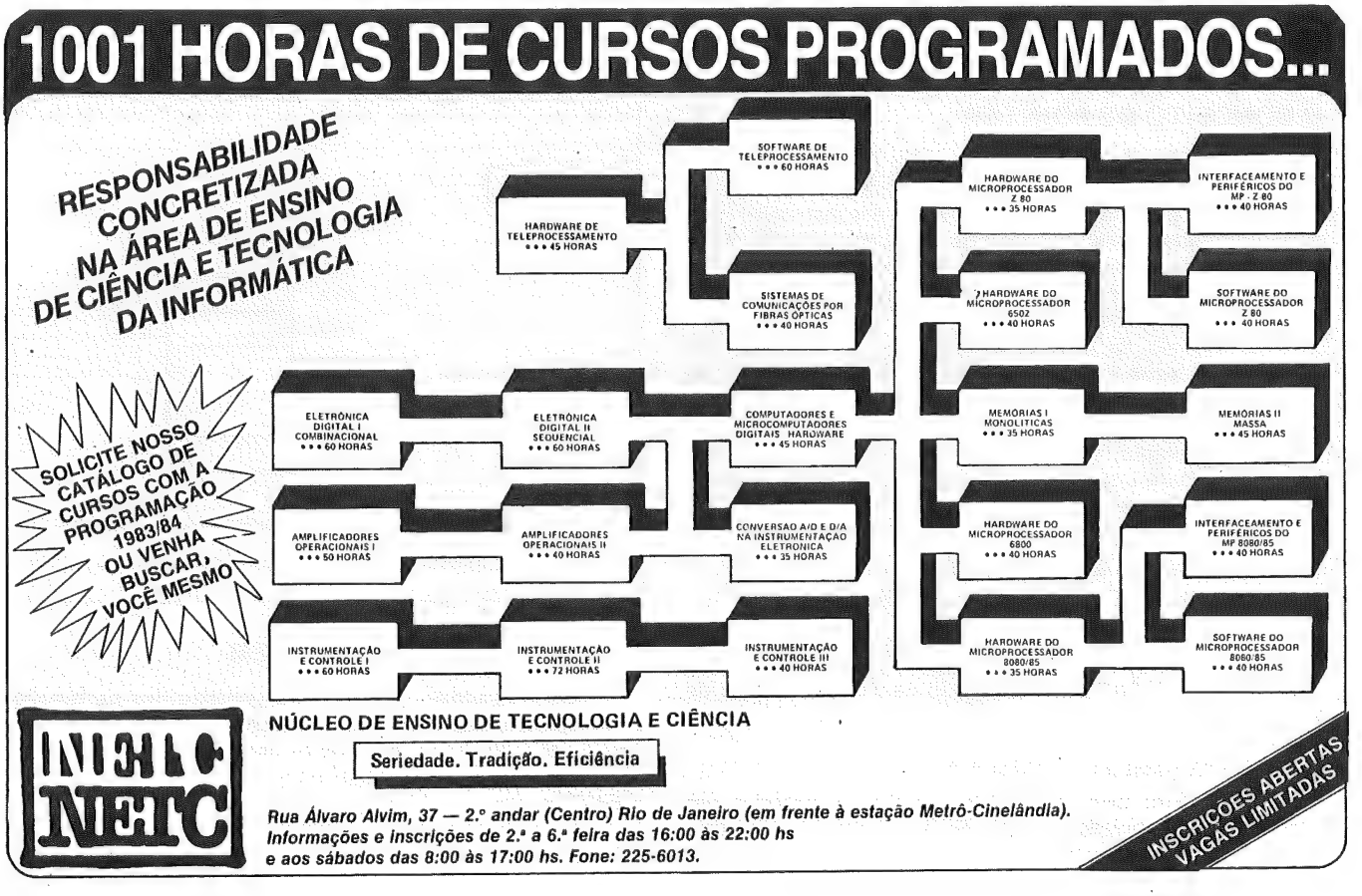
O exemplo mostra como é feita a chamada da função *fun* dentro da função *main*. Parâmetros podem ser passados dentro dos parênteses após o nome da função. Depois do código da função *main* começa o código da função *fun*. Definições de funções não podem estar aninhadas, ou seja, a definição da função *fun* não poderia aparecer dentro da função *main*. Funções podem ser definidas em qualquer ordem, mas a função *main* é normalmente a primeira do programa.

Referências

(1) Learning Programming em C — Thomas Plum — Plum Hall Inc — 1983.

(2) The C Language and Models for Systems Programming — S.C. Johnson and B. W. Kernishan — Byte Magazine — vol. 8, num. 8 — august 1983

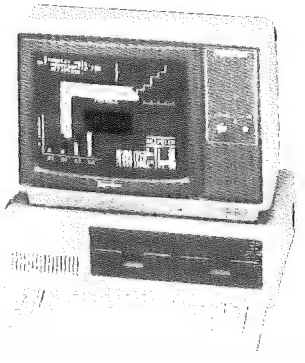
(3) A C Language Primer — James Joyce — idem (2)



DEZEMBRO/83

Interface

mercado de computadores



Como escolher um computador que seja adequado às suas necessidades? Hoje os preços de computadores já são acessíveis, e todos já podem possuir um, inclusive você.

Selecione aqui o seu microcomputador ou periférico e saiba onde adquiri-lo nos mercados do Rio, São Paulo, Belo Horizonte etc.

Características Técnicas & Preços

Novo TK 83

MICRODIGITAL

Mais barato que uma calculadora. Cr$ 149.850,00
preço sujeito a alteração

Ele é Computador Pessoal e Video-Game ao mesmo tempo!

PERSONAL COMPUTER
TK83

O Personal Computer que ensinou computação para mais de 2 milhões de pessoas.

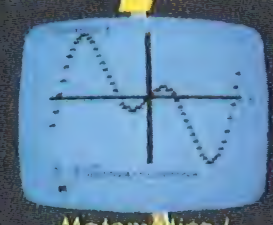
O sucesso do novo Microdigital TK 83 está na sua facilidade de operação: mais de 2 milhões de pessoas em todo o mundo aprenderam a resolver problemas, montar programas e vencer jogos inteligentes usando este equipamento.

Ele trabalha com a Linguagem Basic, a mais rápida de aprender. Pode ter sua memória expandida até 64 K e ser ligado a uma impressora eletro-sensitiva.

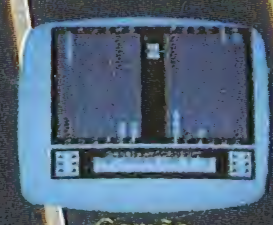
Tem uma quantidade enorme de programas disponíveis no mercado e pode também ser usado com um verdadeiro fliperama de jogos eletrônicos.

Procure já o novo Microdigital TK 83.

É a iniciação mais completa que você pode fazer no mundo da computação.



Matemática I



Gamão

TK 83 Personal Computer

MICRODIGITAL

Microdigital Eletrônica Ltda
Caixa Postal 54088 - CEP 01000 - São Paulo - SP

À venda nas boas casas do ramo, lojas especializadas de fotovideo som, e grandes magazines em: ALAGOAS - Maceió, Palmeira dos Indios, AMAZONAS - Manaus, BAHIA - Salvaodr, CEARÁ - Fortaleza, DISTRITO FEDERAL - Brasília, ESPÍRITO SANTO - Vitória, GOIÁS - Goiânia, MATO GROSSO - Cuiabá, MINAS GERAIS - Belo Horizonte, Divinópoli, Itajuba, Juiz de Fora, Poços de Caldas, São João del Rei, Teófilo Otoni, Uberlândia, Uberaba, Viçosa, PARAÍBA - Campina Grande, PARÁ - Belém, PARANÁ - Curitiba, Londrina, Maringá, PERNAMBUCO - Recife, RIO DE JANEIRO - Campos, Niteroi, Nova Friburgo, Petrópolis, Rezende, Rio de Janeiro, Volta Redonda, RIO GRANDE DO SUL - Bagé, Canoas, Caxias do Sul, Ijuí, Novo Hamburgo, Pelotas, Porto Alegre, Sant'Anna do Livramento, Santiago, Santa Rosa, São Leopoldo, RIO GRANDE DO NORTE - Natal, RONDONIA - Porto Velho, SÃO PAULO - Araraquara, Assis, Avaré, Bauru, Birigui, Botucatu, Campinas, Catanduva, Franca, Guarulhos, Itú, Jacarei, Jaú, Limeira, Lins, Marília, Mogi Guaçu, Mogi das Cruzes, Ourinhos, Piracicaba, Pirassununga, Promissão, Rio Claro, Ribeirão Preto, Santos, Santa Barb. D' Oeste, São Bernardo do Campo, São João da Boa Vista, São Sebastião da Grama, São Carlos, São José do Rio Preto, São José dos Campos, ST° André, São Paulo, Sorocaba, Suzano, Taubaté, SANTA CATARINA - Blumenau, Brusque, Florianópolis, Itajaí, Joinville, SERGIPE - Aracajú. Se você não encontrar este equipamento na sua cidade ligue para (800) 255 8583.

COLEÇÃO

REVISTA **Interface**

CAPA RÍGIDA PARA ENCADERNAÇÃO DE SUA COLEÇÃO

* *2 volumes — encaderna 6 edições por volume*
* *Proteção rígida para todos os seus exemplares*
* *Prática para bibliotecas, laboratórios, cursos, hobbistas, estudantes e professores.*
* *Uma proteção indispensável para sua coleção*
* *Para reservas preencha o cupom abaixo*
* *Também à venda em nosso estande na III Feira Internacional de Informática Parque Anhembi, São Paulo de 17 a 23 de outubro.*

ESTOU INTERESSADO EM ADQUIRIR:

☐ Volume 1 ou volume 2 (Cr$ 2.500,00) ☐ Volume 1 + volume 2 (Cr$ 4.000,00)

GOSTARIA TAMBÉM DE ADQUIRIR OS SEGUINTES EXEMPLARES ATRASADOS PARA COMPLETAR MINHA COLEÇÃO:

Preço unitário: Cr$ 700,00

1 2 3 4 5 6 7 8 9 10 11 12

PARA ISTO ESTOU ENVIANDO CHEQUE NOMINAL À PRODIGT - PROCESSAMENTO, TECNOLOGIA E COMUNICAÇÃO LTDA, ESTRADA DO TINDIBA, 2380 - CEP 22700 - RJ.

NOME: . . . . . .
ENDEREÇO: . . . . . .
EMPRESA: . . . . . .
CARGO: . . . . . TEL: . . . . . .

COMPUTADOR					MEMÓRIA		LINGUAGEM		VÍDEO				
FABRICANTE E MODELO	PREÇO	CPU	RAM	EXPANSÃO DISPONÍVEL	ROM	LINGUAGEM INCLUÍDA	LINGUAGEM DISPONÍVEL	TIPO DE VÍDEO	CARACTERES (LINHA COLUNA)	RESOLUÇÃO GRÁFICA	GRAVADOR CASSETE	DRIVE DE DISCO
MICRODIGITAL TK 82C	*	8 bits Z80 3,25 MHZ	2K	16K 64K	8K	BASIC ASSEMBLER	—	TV P & B	32 x 24	44 x 64	1 gravador 500 BPS	—
MICRODIGITAL TK 83	*	8 bits Z80A 3,25MHZ	2K	16K	8K	BASIC ASSEMBLER	—	TV P & B	32 x 24	44 x 64	1 gravador 500 BPS	—
MICRODIGITAL TK 85	*	8 bits Z80A 3,25 MHZ	16K	48K	10K	BASIC ASSEMBLER	—	TV P & B	32 x 24	44 x 64	1 gravador 500 BPS 4200 BPS (high speed)	—
PROLÓGICA CP 200	*	8 bits Z80A 3,6 MHZ	16K	—	8K	BASIC ASSEMBLER	—	TV P & B	32 x 22	44 x 66	1 gravador 500 BPS	—
PROLÓGICA CP 300	*	8 bits Z80 2 MHZ	48K	—	16K	BASIC ASSEMBLER	—	TV P & B	64 x 16	48 x 128	1 a 2 gravadores 500 BPS, 1500 BPS	1 a 2 mini drive
PROLÓGICA CP 500	*	8 bits Z80 2 MHZ	48K	—	16K	BASIC ASSEMBLER	—	Monitor P & B de 12"	64 x 16	48 x 128	1 a 2 gravadores 500 BPS, 1500 BPS	1 a 2 mini drive
DISMAC D8000	*	8 bits Z80 2 MHZ	16K	48K	16K	BASIC ASSEMBLER	—	Monitor P & B de 12"	64 x 16	48 x 128	1 a 2 gravadores 500 BPS, 1500 BPS	—
CDSE APPLY 300	*	8 bits Z80A 3,25 MHZ	32K	48K	8K	BASIC ASSEMBLER	—	TV P & B e colorida	32 x 24	44 x 64	1 gravador 500 BPS	—
DIGITUS DGT-100	*	8 bits Z80A 2,5 MHZ	16K	48K 64K	14K	BASIC ASSEMBLER	COBOL FORTRAN	TV P & B com adaptação	64 x 16	48 x 128	1 a 2 gravadores 500 BPS 2000 BPS	1 a 4 mini drive
DIGITUS DGT-101	*	8 bits Z80A 2,5 MHZ	16K	64K	14K	BASIC ASSEMBLER	COBOL FORTRAN	TV P & B (vídeo verde) com adaptação	64 x 16	48 x 128	1 a 2 gravadores 500 BPS, 2000 BPS	1 a 4 mini drive
SYSDATA JR	*	8 bits Z80A 1,78 MHZ 3,56 MHZ	16K	48K 62K	14K 2K (opcional)	BASIC ASSEMBLER	BASIC EXPANDIDO	TV P & B ou colorida	64 x 16 32 x 16	48 x 128	1 a 2 gravadores 500 BPS, 1500 BPS	1 a 4 mini drive
UNITRON AP II	*	8 bits 6502 1 MHZ	48K	128K	12K	BASIC ASSEMBLER	BASIC AVANÇADO	TV P & B ou colorida Monitor de 12"	40 x 24	192 x 280	1 gravador 1500 BPS	1 a 4 mini drive
SCOPUS MICROENGENHO	*	8 bits 6502 1 MHZ	48K	128K	12K	BASIC ASSEMBLER	—	TV P & B ou colorida	40 x 24	192 x 280	1 gravador 1500 BPS	1 a 4 mini drive
POLYMAX MAXXI	*	8 bits 6502 1 MHZ	48K	—	12K	BASIC ASSEMBLER	—	TV P & B ou colorida Monitor de 12"	40 x 24	192 x 280	1 gravador 1500 BPS	1 a 4 mini drive
KEMITRON NAJA	*	8 bits Z-80A 3,6 MHZ 4 MHZ	48K	—	16K	BASIC ASSEMBLER	—	TV P & B Monitor de 12"	64 x 16	48 x 128	1 gravador 500 BPS 1500 BPS	1 a 4 mini drive
MAGNEX MG 8065	*	8 bits Z80 6502	64K	128K	10K	BASIC ASSEMBLER	COBOL FORTRAN	TV P & B ou colorida Monitor de 12"	40 x 24 80 x 24	192 x 280	1 gravador 1500 BPS	1 a 2 mini drive ou 1 a 2 drive de 8"

*** Consulte o seu revendedor pois o preço de um computador depende da sua configuração, onde estão envolvidos interfaces, periféricos, cabos de conexão e expansões de memória.**

LISTA GERAL DOS FORNECEDORES

Manaus: PRODADOS (092) 234-1045
Salvador: OFICCINA (071) 248-6666 R. 268
SISPROL (071) 247-8951
Fortaleza: ABACO (085) 226-4922
COMPUT (085) 224-0544
Vitória: LOGDATA (027) 222-5811
Brasília: COMPUSHOW (061) 273-2128
CINE FOTO GB (061) 242-6344
SBM (061) 226-1523
TELESERVICE (061) 226-0133
Goiânia: ASSISTE (062) 224-7098
São Luiz: DIGITOS (098) 222-6691
Belo Horizonte: BYTE SHOP (031) 223-6947
COMPUCITY (031) 226-6336
COMPUTRONIX (031) 225-3305
MINAS DIGITAL (031) 201-7555
KEMITRON (031) 225-0644
Muriné: REGIS CINE FOTO SOM (032) 721-1593
Belém: BELDATA (091) 228-0011
COMPUTRON (091) 222-5122
Curitiba: COMPU-INN-SYSTEM (041) 243-1731
COMPUSHOP/COMPUSTORE (041) 232-1750
Cascavel: MICROLINE (0452) 23-1538
João Pessoa: MEDUZA (083) 221-6743
Recife: OFICCINA (081) 326-9318
SOTEMAQ (081) 231-6796
Terezina: MARGHUS (086) 222-0186
Rio de Janeiro: CLAPPY (021) 253-3395
BITS E BYTES (021) 322-1960
IPANEMA MICRO (021) 259-1516
ELETRODATA (021) 288-2650
KRISTIAN (021) 252-9057
MICROHOUSE (021) 294-6248
MICROSHOW (021) 264-5512

PERIFÉRICOS		SOFTWARE		COMENTÁRIOS
DISQUETE	INTERFACES	SISTEMA OPERACIONAL	SOFTWARE APLICATIVO	
—	Impressora Eletro-Sensível e Joystick	Monitor	Jogos, gráficos, aplicações domésticas e comerciais para pequenas empresas. Grande quantidade de software disponível no mercado.	Compatível com o Sinclair ZX81.
—	Impressora Eletro-Sensível e Joystick	Monitor	Jogos, gráficos, aplicações domésticas e comerciais para pequenas empresas. Grande quantidade de software disponível no mercado.	Compatível com o TK82-C, "Design' mais compacto, utilizando componentes LSI e teclado de membrana em dupla camada.
—	Impressora Eletro-Sensível e Joystick	Monitor	Jogos, gráficos, aplicações domésticas e comerciais para pequenas empresas. Grande quantidade de software disponível no mercado.	Teclado em alto-relevo alfanumérico, com 40 teclas e 154 funções.
—	—	Monitor	Jogos, gráficos, aplicações domésticas e comerciais para pequenas empresas. Grande quantidade de software disponível no mercado.	Teclado em alto-relevo alfanumérico, com 40 teclas e 154 funções.
5 1/4"	Controlador de disco, Impressora paralela e RS 232	Monitor D.O.S. 500	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível em software e hardware com o CP 500, sendo composto de módulos que podem ser adquiridos de acordo com as suas necessidades.
5 1/4"	Controlador de disco, Impressora paralela e RS 232	D.O.S. 500	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível com o TRS 80 modelo III.
—	Impressora paralela, RS 232 e S100	Monitor	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	O fabricante não tem disponível a interface de disco até o momento.
—	RS 232, Joystick	Monitor	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Lançamento recente no mercado de Informática pela CDSE Microcomputadores, à Estrada do Galeão, 11 s/202 RJ — Tel.: (021) 396-4264.
5 1/4"	Controlador de disco, Impressora paralela, RS 232 e sintetizador de voz	NDOS DGP/M	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível com o TRS 80 modelo I porém com todas as interfaces desenvolvidas no Brasil.
5 1/4"	Controlador de disco, Impressora paralela, RS 232 e sintetizador de voz	DGP/M (Compatível com o CP/M 2.2)	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	O sistema é fornecido com todos os cabos de conexões, interfaces de drive de disco e impressora e duas unidades de drive de disco.
5 1/4"	Controlador de disco, Impressora paralela e RS 232	D.O.S.	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	O JR é compatível com o TRS80,
5 1/4"	Controlador de disco, Impressora paralela, RS 232 e Joystick	D.O.S. 3.3	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível com o APPLE II (D.O.S. 3.3).
5 1/4"	Controlador de disco, Impressora paralela, RS 232 e Joystick	D.O.S. 3.3	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível com o APPLE II (D.O.S. 3.3).
5 1/4"	Controlador de disco, Impressora paralela e Serial, RS 232 e Joystick	D.O.S. 3.3	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível com o APPLE II (D.O.S. 3.3).
5 1/4"	Controlador de disco, Impressora paralela e RS 232	TRS D.O.S.	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível com o TRS-80 Modelo III.
5 1/4" e 8"	Controlador de disco, Impressora Serial, caneta de luz e Joystick	D.O.S. 3.3 CPM 2.2	Gráficos, estatística, cálculos matemáticos, programas educacionais, aplicações profissionais e comerciais para pequenas e médias empresas.	Compatível com o D.O.S. 3.3 e CPM 2.2.

LISTA GERAL DOS FORNECEDORES

GARSON (021) 252-2050
NASAJON (021) 263-1241
MICRO KIT (021) 267-8291
LHM (021) 262-5437
MICROEQUIPO (021) 220-5059
ELDORADO (021) 227-0791
COMPUTIQUE (021) 267-1093
MICROCENTER (021) 264-0143
TESBI (021) 591-3297

Natal: ECONSULT (084) 222-3212

Porto Alegre: ADVANCING (0512) 26-1194
DIGITAL (0512) 24-1411
INFORMATIQUE (0512) 22-4198
MICROSIS (0512) 34-0660

Caxias: MICROSUL (054) 221-8301

Frederico Westphalen: ELETRODATA (055) 344-1550

Novo Hamburgo: MICROMEGA (0512) 93-4721

São Paulo: COMPUSHOP (011) 212-9004
CEI-SERVIMEC (011) 222-1511
COPEC (011) 67-0063
MICROSHOP (011) 282-2105
IMARÉS (011) 61-0946
COMPUTERLAND (011) 258-3954/258-1573
COMPUTIQUE (011) 231-3922
MAGNEX (011) 570-2872/549-2232

Barretos: INTEC (0173) 22-6411

Piracicaba: PALMA (0194) 33-1470

Ribeirão Preto: CONSIMAF (016) 625-5924
COMPUSYS (016) 635-1195
DECISA (016) 625-5926
MEMOCARDS (016) 636-0586

São Joaquim da Barra: ITUVEMAQ (016) 728-2472

São José dos Campos: SISCOM (0123) 23-3752

São José do Rio Preto: MICROM (0172) 32-0600

Florianópolis: CSTRO COMPUTADORES (0482) 23-0491

Aracaju: MICROMUNDO (079) 224-1310

Campinas: COMPUTIQUE (0192) 32-6322

Poços de Caldas: COMPUTIQUE (035) 721-5810

balcão

CLASSIFICADOS DO LEITOR: REMETA SEU ANÚNCIO PARA REVISTA INTERFACE – MERCADO DE COMPUTADORES – ESTRADA DO TINDIBA, 2380. JACAREPAGUÁ – CEP 22700 – RIO DE JANEIRO – RJ.

Troco ou vendo programas para os micros CP-200, TK 82/85 e Sinclair Spectrun. Possuo o Galactica, Mazogs, Hipper, Kong e muitos outros. Eduardo M. A. de Medeiros. Rua Eliseu Guilherme, nº 1076 – Sumaré – Ribeirão Preto-SP – 14100, Tel. (016) 625-5796.

Vendo ou troco programas para TK82-C, NE, Sinclair e similares. Possuo Defender, Crazy Kong, Redalert e muitos outros. Aos interessados contatar Vânia pelo tel. 289-9066.

Jogo de xadrez com microcomputador 7 níveis de dificuldade e várias opções de jogo. É similar ao *Byte 300*. Falar com Antonio. Tel. 263-3171 e 254-6815. Rua Ribeiro Guimarães, nº 80/1706 – Rio de Janeiro-RJ.

Vendo ou troco programas para microcomputadores TK-82, NEZ-8000, SIN-CLAIR ZX-81, TIMEX 1000 e CP-200. De preferência troco. Favor escrever para Adelson de Carvalho Acioly Jr. Rua da Nola, 465/102-B, Recife-PE – 50000 – Tel. (081) 241-3816.

Expande-se hardware de DGT-100 e TK-85, tornando-os aplicáveis ao uso científico e tecnológico. Caixa Postal 14717 – Ipanema-Rio.

Vendo 10 programas em linguagem de máquina para microcomputadores TK82-C, TK85, CP-200, NEZ8000, SINCLAIR ou similares, por Cr$ 30.000,00. Vem com fita SCOTCH e você é quem escolhe. Tenho vários, como: KRAZY, KONG, XADREZ, PAL-MAN, LABIRINTO e muitos outros. Vendo também programa que transforma seu TK82-C ou CP-200 em HI-SPEED (4.200 bauds). Entrar em contato com José Antônio, Rua Conde de Bonfim, nº 912/501 – Tijuca-RJ - Tel. (021) 258-4537 - 20530.

Vendo ou troco programas para TK82-C, NEZ8000, SINCLAIR ZX81 e CP-200. Tenho variedades de jogos gráficos, aplicativos, matemáticos etc. Tratar com José Antonio, telefone 262-2064 (horário comercial inclusive aos sábados).

SOFTSYSTEM – Software importado para TK 82/85, NEZ, CP-200, ZX-81 etc. Linguagem de máquina; inéditas. Solicite catálogo para Caixa Postal 02 – Poá -SP – 08550.

TK-85 – Folha de pagamento, agenda de telefones, controle de estoques, contas a pagar/receber, fluxo de caixa, mala direta, processador texto, controle bancário, cadastro clientes, contabilidade doméstica e orçamento doméstico. Contatar Bete pelo tel. (011) 284-5635.

Vendo 1 (uma) fita com os seguintes programas: Labirinto, Monstro das Trevas, Fugitivo, Demolidor, Agenda e Orçamento Doméstico; para TK-82C, TK-85, CP-200 etc, por Cr$ 10.000,00. Vicente Quartieri, Rua Ipê, nº 110 - São Vicente-SP – Tel. 67-4805. Atendo por reembolso ou vale postal.

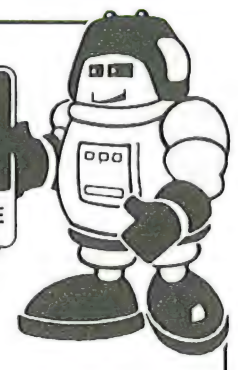
GRANDE CIRCUITO – Boletim para usuários do TK/NE/CP-200. Anúncios grátis. Envie selo para resposta e receba um exemplar com carteirinha de sócio inteiramente grátis. Caixa Postal 28, Piraí-RJ – 27200.

HP – 85 – MICROCOMPUTADOR CIENTÍFICO – Ideal para aplicações nas áreas de engenharia, matemática, estatística etc. Expandido para 32Kb, impressora e vídeo acoplados, com pacotes financeiro, estatísticos I, II e III, Visicalc Plus e maleta original.

NOVO

Agora você dispõe da mais avançada e ~~sofisticada~~ tecnologia em software para microcomputadores, livrando você dos inconvenientes gerados por produções amadoras ou de origem duvidosa.

CIBERNE
O SOFTWARE QUE VOCÊ MERECE



VEJA O QUE CIBERNE LHE OFERECE

- Diversificada linha de programas
- Novos lançamentos periodicamente
- Pacotes econômicos
- Gravação profissional
- Embalagem inviolável
- Garantia Total

PROGRAMAS EM FITA PARA TK-82, 83, 85, CP-200 E COMPATÍVEIS

Com o exclusivo FLASH-SISTEM que permite um carregamento 6 vezes mais rápido que o normal, sem qualquer modificação no equipamento.

PROCURE EM SEU REVENDEDOR:

- CASCA A COBRA/CENTOPÉIA – Na mesma fita, dois excitantes jogos que são um desafio à sua habilidade e raciocínio.
- FUNGOS MUTANTES/CRAZY KONG – Perigosos vegetalóides atacam uma colônia de humanos. E mais, o clássico Kong agora melhorado e com 3 fases.
- NAVE MÃE/GALÁCTICA – Desta vez a invasão de extraterrestres vai deixar você arrepiado de emoção colocando à prova sua perícia e inteligência.
- DEFENSOR/MAZOGS – Você dirige a última nave que tenta defender uma civilização perdida. E o mais fantástico dos caça-ao-tesouro.
- GUERRA AÉREA/ALERTA VERMELHO – Duas emocionantes aventuras aéreas onde você pilota um caça ou um poderoso bombardeiro.
- ROT-I PLUS – Incremente seus programas com este sensacional sistema operacional gráfico, uma nova e mais poderosa versão de ROT-I.

E PARA BREVE:
- O MERCADOR
- STARQUEST
- ZARAKS
- ESTRELA NEGRA
- CRISTAL MÁGICO
- GUERRILHA CÓSMICA
- ALTA RESOLUÇÃO
- ORÇA I (Orçamento Doméstico)
- ROT II (Compilador Ass/Desas)
- ARQ I (Arquivo de Dados)
- CASH-FLOW

E MAIS:
- VASTA LINHA DE PROGRAMAS PARA DIGITUS, CP-300 E 500, APPLE E MUITOS OUTROS.

EM TODO O BRASIL NAS MELHORES LOJAS DO RAMO.

DESCONTOS ESPECIAIS PARA REVENDEDORES.

Informações, Distribuição e Vendas:

JVA – MICROCOMPUTADORES LTDA.
Av. Treze de Maio, 23 – Grupo 1519
CEP 20.031 – Rio de Janeiro - RJ
Tel.: (021) 262-6968

BITS & BYTES
COMPUTADORES

- VENDAS
- PROGRAMAS
- DISKETTS
- FITAS
- SERVIÇOS
- CURSOS DE BASIC
- FORMULÁRIOS
- ASS. TÉCNICA (AUTORIZADA PROLÓGICA)

CONSERTOS EM 24 HORAS (COM GARANTIA) PARA O CP-500 e DGT-100

EM SÃO CONRADO
Estrada da Gávea, 642 — Lj. B
TEL.: 322-1960

LOJA MICRO-KIT
TUDO SOBRE MICROCOMPUTADOR

FAÇA AGORA A SUA ENCOMENDA

LANÇAMENTO DO LIVRO "CURSO DE BASIC" — VOL. I — EDIÇÃO PRÓPRIA

CURSOS

Basic p/adultos e crianças, com método próprio comprovadamente eficiente; Professores c/mestrado em ENGENHARIA DE SISTEMAS, mais de 20 cursos aplicados. Turmas pequenas, aulas práticas com MICROCOMPUTADOR.

VENDA DE MICROCOMPUTADOR

Unitron AP II, Digitus, TK e CP 200.
Financiamento em até 24 meses.

PROGRAMAS

Comerciais e Jogos p/APPLE, Unitron, Polymax, Digitus TK e CP 200.

SUPRIMENTOS

Disquetes, Caixa p/Disquetes, Formulários Contínuos etc.

VENDA DE LIVROS E REVISTAS

Despachamos para todo o Brasil.

Rua Visconde de Pirajá, 303 S/Loja 210
Tels. (021) 267-8291 521-4638
CEP 22410 — Rio de Janeiro
Rua Visconde de Pirajá, 365 — Sobreloja 209
Ipanema

ASSEL
ASSISTÊNCIA ELETRÔNICA LTDA.

ASSISTÊNCIA TÉCNICA
CALCULADORAS ELETRÔNICAS
MICROCOMPUTADORES E ACESSÓRIOS
AUTORIZADO:
TEXAS E DISMAC

RUA DA LAPA, 107 - 10º AND. - RIO - RJ.
TELS.: 222-7137 E 222-2278

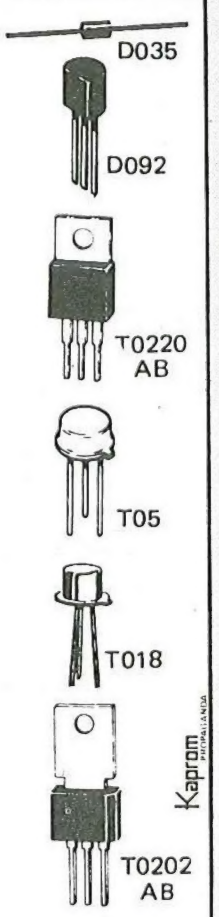
AV. MINISTRO EDGARD ROMERO, 81
- MADUREIRA - SOBRELOJA 307

A CERTEZA DE UM BOM NEGÓCIO

TIPO	CORRENTE DIRETA IF (RMS) A (max)	TENSÃO REVERSA VRRM V (min)	ENCAPSULAMENTO
RETIFICADORES CONTROLADOS DE SILÍCIO (SCR)			
S2001MS2	1,6	200	T05
S6003L	3	600	T0220AB
T106D1	4	400	T0202AB
T107D1	4	400	T0202AB
S2006L	6	200	T0220AB
S4015L	15	400	T0220AB
RETIFICADORES CONTROLADOS DE SILÍCIO BIDIRECIONAIS			
Q200E3	0,8	200 a 400	T092
Q2004F31	4	200 a 400	T0202AB
Q2008L4	8	200 a 600	T0220AB
Q2010L4	10	200 a 600	T0220AB
Q2015L5	15	200 a 600	T0220AB
TRIAC LÓGICO			
L4004F51	4	400	T0202AB
QUADRACS (TRIACS COM DIACS INCORPORADOS)			
Q4004FT1	4	400	T0202AB
Q4006LT	6	400	T0220AB
Q2008FT1	8	200	T0202AB

(TRIAC)

TECCOR ELECTRONICS INC



Kaprom

SCR ATIVADO POR LUZ

PR 30 — Tensão Bloqueio Direta Reversa 30 volts (min)
Corrente Bloqueio Direta Reversa 25 uA (max) (T018)

DISPARADOR BILATERAL DE SILÍCIO (DIAC)

HT32 — Vbo = 27/32t/37, 1,5A para pulso de 10uS de duração (D035)

Teleimport Eletrônica Ltda.
Rua Sta. Ifigênia, 402, 8/10º andar - CEP 01207 - São Paulo
Fone: 222-2122 - Telex (011) 24888 TLIM-BR
(Solicite nosso catálogo geral de componentes)

OFERTAS Kristian

MICROCOMPUTADORES

DGT-100 Cr$ 220.000, x 3 — Grátis 18 JOGOS
CP-500 Cr$ 790.000, — Grátis 18 JOGOS
CP-200 Cr$ 100.000, x 2 — Grátis 18 JOGOS
TK82-C Cr$ 49.925, x 2 — Grátis 10 JOGOS
ainda UNITRON Ap II, Mem 64K, Impressoras, etc...
(Preços sujeitos a modificações)

PROGRAMAS PRONTOS EM FITAS

JOGOS — TK ou CP

- VISITA AO CA$$INO
- MIDWAY
- PASSAGEM PARA O INFINITO
- 10 JOGOS EXCITANTES PARA 1K
- 2ª DIMENSÃO
- JORNADA NAS ESTRELAS

E MUITO MAIS!

JOGOS: — DGT-100 OU CP-500

- SCARFMAN
- PENETRATOR
- SUPER-NOVA
- VIAGEM A VALKYRIA
- ASILO 1
- AVENTURAS
- DEFENSE COMMAND

E MUITO MAIS!

LEASING E CRÉDITO DIRETO!

LITERATURA

- MICRO-SISTEMAS
- INTERFACE
- JORNAL TK-CP
- IMPORTADOS

+ CURSOS DE BASIC GRÁTIS
NA COMPRA DE QUALQUER MICRO

DESPACHAMOS PARA TODO O BRASIL!

APLICATIVOS

- CONTROLE DE ESTOQUE
- CONTAS A PAGAR/RECEBER
- MALA DIRETA/CADASTRO
- FOLHA DE PAGAMENTO
- VÍDEO-CLUBES
- ESTATÍSTICOS
- SOFTWARE SOB ENCOMENDA

Kristian ELETRÔNICA LTDA.
Rua da Lapa, 120 Gr. 505
Rio de Janeiro - RJ
Tel.: (021) 252-9057

ACESSÓRIOS PARA COMPUTAÇÃO

Em aço, com a qualidade **WALNE!**

- ESTANTES PARA FITAS EM ESTOJO OU TAPE-SEAL.
- ARMÁRIOS E ARQUIVOS PARA DISCOS (panelas) e DISKETTS.
- CARROS e ESTOJOS para transporte DISCOS e FITAS.
- ARMÁRIOS e ESTANTES de MATERIAL.
- MESAS, CADEIRAS cestos lixões.
- ARQUIVOS e CARROS para LISTAGENS.
- ARQUIVO especiais para MICROFILME.
- Sistema de arquivo visível "VR-WALNE" p. Controle Fitoteca.

(Solicite o catálogo da linha de Acessório de Computação)

Tudo para computação menos o computador, fita, o disco e o cartão.

WALNE EQUIPAMENTOS E SISTEMAS LTDA.

DEPTO. VENDAS: RUA RODRIGO SILVA, 18
2º AND. TEL. 224-2200 – CEP 20.011 – RJ.

MC INFORMATICA

TUDO EM MICROCOMPUTADORES

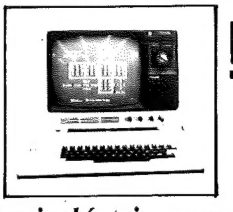
CURSOS	MICROS
BASIC	
BASIC AVANÇADO	TK83 CP-200
ASSEMBLER	TK85 CP-300
COBOL	TK2000 CP-500
PASCAL	DGT-1000 MAXXI
FORTRAN	MICRO ENGENHO 1 E 2
COM PRÁTICA NOS MICROS	

MC-SOFT
PROGRAMA PARA TODOS OS MICROS

Microcenter Informática Ltda
Rua Conde de Bonfim, 229 – Lojas 310 e 312
Tijuca – Rio de Janeiro-RJ.
CEP 20520 – Tel. (021) 264-0143/228-0593

INSCREVA-SE JÁ

INFORMÁTICA



INFORMÁTICA

A INFORMÁTICA é a indústria que mais cresce no mundo
a que paga os melhores salários
e a única carente de técnicos especializados

FORMAÇÃO DE TÉCNICOS ESPECIALIZADOS

PROGRAMAÇÃO E MANUTENÇÃO DE MICROCOMPUTADORES

PRÉ-REQUISITO : 2º GRAU COMPLETO
DURAÇÃO : 920 HORAS
HORÁRIO : 19.00 às 22.30 HORAS
2ª a 6ª feira

Estágio Supervisionado
Certificado de Estabelecimento do Sistema Oficial de Ensino.

CURSOS EXTRA-CURRICULARES

- PROGRAMAÇÃO BASIC
- BASIC AVANÇADO
- LÓGICA DIGITAL (Grátis: Treinador Lógico)
- MICROPROCESSADORES – 8080/85 – Z-80
- SISTEMA OPERACIONAL CP/M
- COBOL

Visite nossos laboratórios

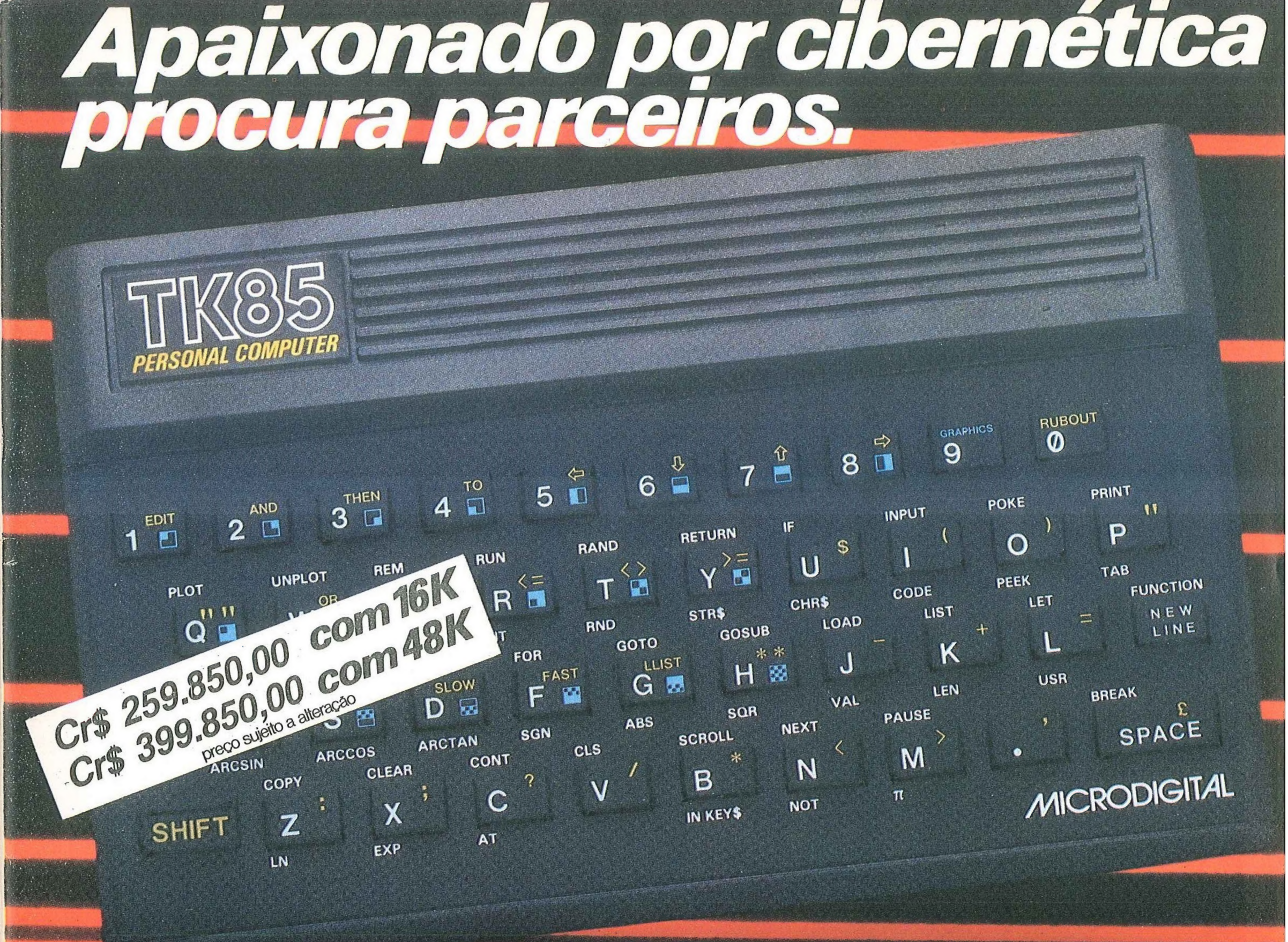
CENTRO EDUCACIONAL CARVALHO DE MENDONÇA

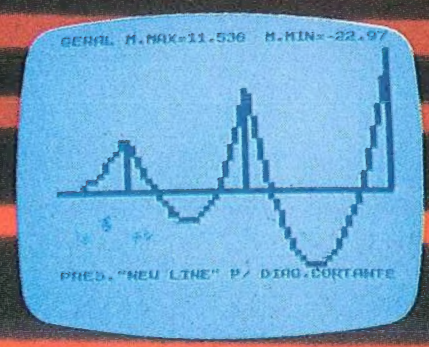
ENGEMICRO
ENGENHARIA EDUCACIONAL DE MICROPROCESSAMENTO

Rua Evaristo da Veiga, 20 - Tel. 220-8820/220-7009

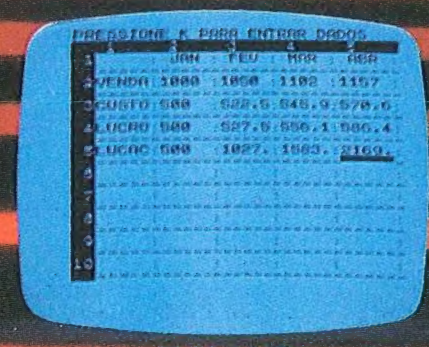
CERTIFICADO: REGISTRO NO SISTEMA OFICIAL DE ENSINO

Apaixonado por cibernética procura parceiros.

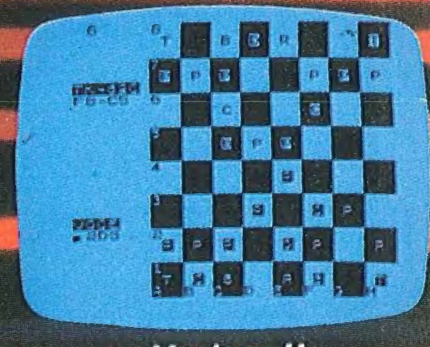




Viga Contínua



T-KALC



Xadrez II

Computador Pessoal TK 85

MICRODIGITAL

Tenho grandes recursos sou compacto e muito fácil de usar (deve ser por isso que eu sou o computador pessoal mais conhecido do mercado brasileiro).

Ajudo você a resolver seus assuntos profissionais e domésticos. E posso jogar com você centenas de jogos.

E tenho também características muito avançadas: teclado tipo máquina de escrever, high-speed, e a função verify (para sua segurança ao guardar seus programas e dados em fita cassete).

Venha me conhecer. Eu posso ser o seu parceiro ideal por muito tempo.

Outras características técnicas importantes:

- Linguagens Basic e Assembler
- 16 ou 48 K bytes de memória RAM
- 10 K bytes de ROM
- 40 teclas com 160 funções
- Gravação de programas em fita cassete comum
- Input e output de dados
- Video: TV P&B ou cores
- Funções especiais para gravação em High-speed (4200 BAUDS)
- Possibilidade de acoplar Joystick
- Possibilidade de acoplar impressora

Microdigital Eletrônica Ltda.
Caixa Postal - 54088 - CEP. 01000 - São Paulo - SP

À venda nas boas casas do ramo, lojas especializadas de foto-video-som, e grandes magazines em: ALAGOAS - Maceió, Palmeira dos Indios, AMAZONAS - Manaus, BAHIA - Salvador, CEARÁ -Fortaleza, DISTRITO FEDERAL - Brasília, ESPÍRITO SANTO -Vitória, GOIÁS - Goiânia, MATO GROSSO - Cuiabá, MINAS GERAIS -Belo Horizonte, Divinópolis, Itajubá, Juiz de Fora, Poços de Caldas, São João Del Rei, Teófilo Otoni, Uberlândia, Uberaba, Viçosa, PARAÍBA - Campina Grande, PARÁ - Belém, PARANÁ - Curitiba, Londrina, Maringá, PERNAMBUCO - Recife, RIO DE JANEIRO - Campos, Niterói, Nova Friburgo, Petrópolis, Resende, Rio de Janeiro, Volta Redonda, RIO GRANDE DO SUL - Bagé, Canoas, Caxias do Sul, Ijuí, Nova Hamburgo, Pelotas, Porto Alegre, Sant'Anna do Livramento, Santiago, Santa Rosa, São Leopoldo, RIO GRANDE DO NORTE - Natal, RONDONIA - Porto Velho, SÃO PAULO - Araraquara, Assis, Avaré, Bauru, Birigui, Botucatu, Campinas, Catanduva, Franca, Guarulhos, Itú, Jacarei, Jaú, Limeira, Lins, Marilia, Mogi Guaçu, Mogi das Cruzes, Ourinhos, Piracicaba, Pirassununga, Promissão, Rio Claro, Ribeirão Preto, Santos, Santa Barb. D'Oeste, São Bernardo do Campo, São João da Boa Vista, São Sebas. da Grama, São Carlos, São José do Rio Preto, São José dos Campos, St.º André, São Paulo, Sorocaba, Suzano, Taubaté, SANTA CATARINA - Blumenau, Brusque, Florianópolis, Itajaí, Joinville.

Se você não encontrar este equipamento na sua cidade ligue para (800) 255.8583.

NOVO CP 300 PROLÓGICA.

O pequeno grande micro.

Compatível com programas em fita cassete ou em disco.

Agora, na hora de escolher entre um microcomputador pessoal simples, de fácil manejo e um sofisticado microcomputador profissional, você pode ficar com os dois.

Porque chegou o novo CP 300 Prológica.

O novo CP 300 tem preço de microcomputador pequeno. Mas memória de microcomputador grande.

Ele já nasceu com 64 kbytes de memória interna com possibilidade de expansão de memória externa para até quase 1 megabyte.

Pode ser acoplado a uma impressora.

E tem um teclado profissional, que dá ao CP 300 uma versatilidade incrível. Ele pode ser utilizado com programas de fita cassete, da mesma maneira que com programas em disco.

O único na sua faixa que já nasce com 64 kbytes de memória.

Permite conexão telefônica.

Pode ser ligado ao seu aparelho de TV, da mesma forma que no terminal de vídeo de uma grande empresa.

Com o CP 300 você pode fazer conexões telefônicas para coleta de dados, se utilizar de uma impressora e ainda dispor de todos os programas existentes para o CP 500 ou o TRS-80 americano. E o que é melhor: você estará apto a operar qualquer outro sistema de microcomputador.

Pode ser ligado a um televisor comum ou a um sofisticado terminal de vídeo.

Nenhum outro microcomputador pessoal na sua faixa tem tantas possibilidades de expansão ou desempenho igual.

CP 300 Prológica.

Os outros não fazem o que ele faz, pelo preço que ele cobra.

PROLOGICA
microcomputadores

Av. Eng.º Luis Carlos Berrini, 1168 - SP

RESERVA DE MERCADO

DEFESA DOS VALORES NACIONAIS

AM
Manaus - 234-1045
• BA-Salvador - 247-8951
• CE-Fortaleza - 226-0871 - 244-2448
• DF-Brasília - 226-1523 - 225-4534 • ES-Vila Velha
229-1387 - Vitória - 222-5811 • GO-Goiânia - 224-7098 • MT
Cuiabá - 321-2307 • MS-Campo Grande - 383-1270 - Dourados - 421-1052
• MG-Belo Horizonte - 227-0881 - Betim - 531-3806 - Cel. Fabriciano - 841-3400 - Juiz
de Fora - 212-9075 - Uberlândia - 235-1099 • PA-Belém - 228-0011 • PR-Cascavel - 23-1538 - Curitiba - 224-5616 - 224-3422 - Foz do Iguaçu - 73-3734 - Londrina - 23-0065 • PE-Recife - 221-0142 • PI-Teresina
222-0186 • RJ-Campos - 22-3714 - Rio de Janeiro - 264-5797 - 253-3395 - 252-2050 • RN-Natal - 222-3212 • RS-Caxias do
Sul - 221-3516 - Pelotas - 22-9918 - Porto Alegre - 22-4800 - 24-0311 - Santa Rosa - 512-1399 • RO-Porto Velho - 221-2656 • SP
Barretos - 22-6411 - Campinas - 2-4483 - Jundiaí - 434-0222 - Marília - 33-5099 - Mogi das Cruzes - 469-6640 - Piracicaba - 33-1470 - Ribeirão
Preto - 625-5926 - 635-1195 - São Joaquim da Barra - 728-2472 - São José dos Campos - 22-7311 - 22-4740 - São José do Rio Preto - 32-2842 - Santos - 33-2230
Sorocaba - 33-7794 • SC-Blumenau - 22-6277 - Chapecó - 22-0001 - Criciúma - 33-2604 - Florianópolis - 22-9622 - Joinvile - 33-7520 • SE-Aracajú - 224-1310.

Solicite demonstração nos principais magazines.